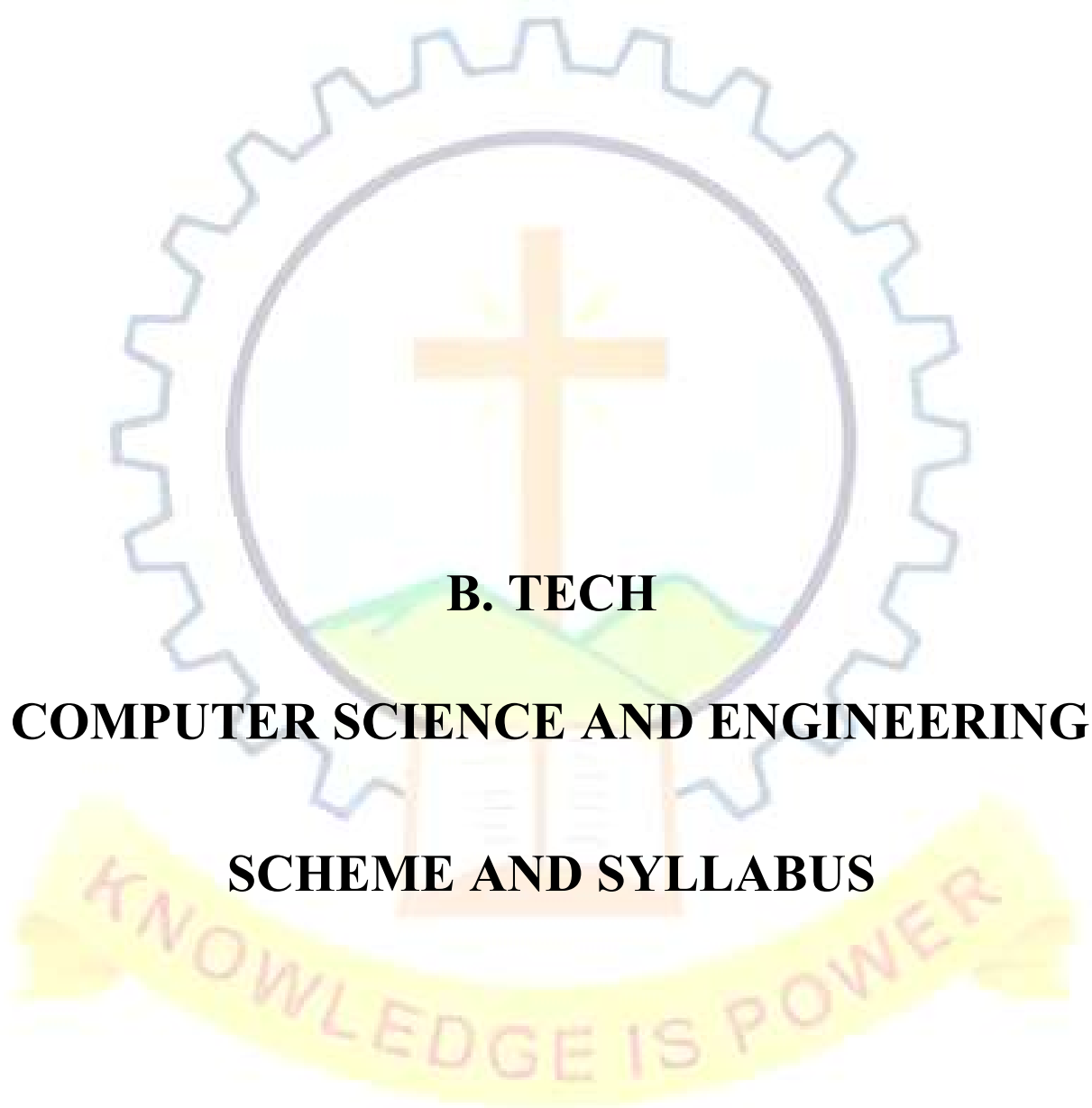


MAR ATHANASIUS COLLEGE OF ENGINEERING
(Government Aided and Autonomous)

Kothamangalam-686666



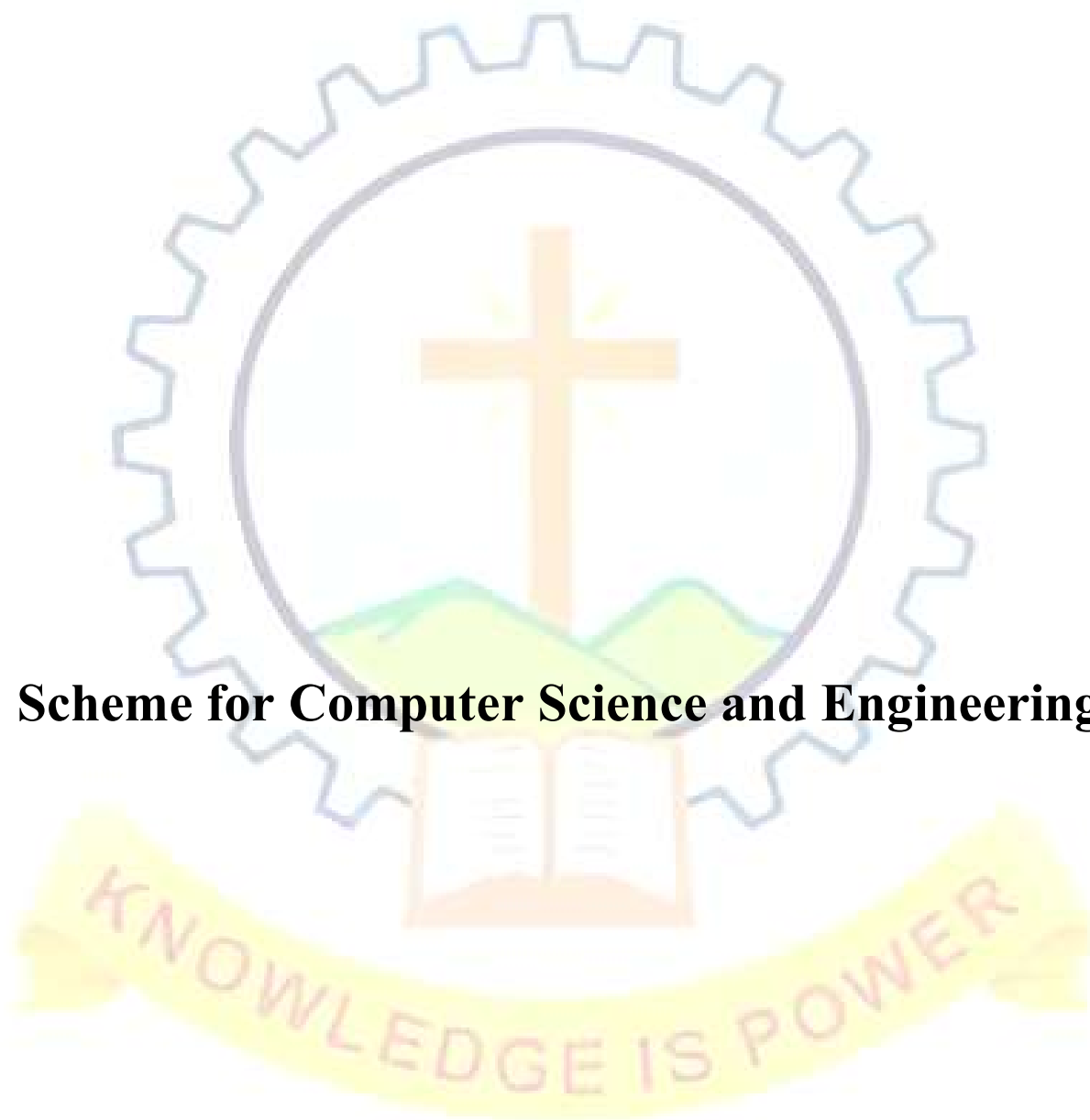
B. TECH

COMPUTER SCIENCE AND ENGINEERING

SCHEME AND SYLLABUS

MAR ATHANASIUS COLLEGE OF ENGINEERING
(Government Aided and Autonomous)
Kothamangalam-686666

Scheme for Computer Science and Engineering



SEMESTER 1

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24MA1T01	LINEAR ALGEBRA AND MULTIVARIABLE CALCULUS	3-1-0-5	4	4
B	B24ES1T01A	PROBLEM SOLVING & PROGRAMMING TECHNIQUES (A)	2-1-0-4	3	3
C	B24PH1T01A	ENGINEERING PHYSICS (A)	2-1-0-4	3	3
D	B24CY1T01A	ENGINEERING CHEMISTRY (A)	2-1-0-4	3	3
E	B24ES1T02	BASICS OF ELECTRICAL & ELECTRONICS ENGINEERING	2-2-0-5	4	4
F	B24ES1L02	BASIC ELECTRICAL AND ELECTRONICS WORKSHOP	0-0-2-1	2	1
G	B24ES1L01A	PROGRAMMING LAB (A)	0-0-3-2	3	2
H	B24PC1L01A	PHYSICS (A) & CHEMISTRY (A) LAB	0-0-2-2	2	1
I	B24MC1T01	LIFE SKILLS	1-0-1-2	2	P/F
J	B24MC1T02	DESIGN THINKING	1-1-0-2	2	P/F
K	B24MC1L01	YOGA AND SPORTS	0-1-1-1	2	P/F
TOTAL				30	21

SEMESTER 2

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24MA1T02	ORDINARY DIFFERENTIAL EQUATIONS AND TRANSFORMS	3-1-0-5	4	4
B	B24ES1T03A	COMPUTER AIDED ENGINEERING GRAPHICS (A)	2-0-2-4	4	3
C	B24CS1T01	LOGIC SYSTEM DESIGN	3-1-0-5	4	4
D	B24CS1T02	INDUSTRIAL PROGRAMMING	2-1-0-4	3	3
E	B24CS1T03	OBJECT ORIENTED PROGRAMMING	3-1-0-5	4	4
G	B24CS1L01	INDUSTRIAL PROGRAMMING LAB	0-0-3-2	3	2
H	B24CS1L02	OBJECT ORIENTED PROGRAMMING LAB	0-0-3-2	3	2
I	B24MC1T03	PROFESSIONAL COMMUNICATION & ETHICS	2-0-1-4	3	P/F
J	B24MC1L02	IDEA LAB	0-0-2-2	2	P/F
TOTAL				30	22

SEMESTER 3

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24MA2T03C	DISCRETE MATHEMATICS	3-1-0-5	4	4
B	B24CS2T01	DATA STRUCTURES	3-1-0-5	4	4
C	B24CS2T02	COMPUTER ORGANIZATION AND ARCHITECTURE	3-1-0-5	4	4
D	B24CS2T03	COMPUTER NETWORKS	2-1-0-4	3	3
E	B24HU2T01	BUSINESS ECONOMICS AND FINANCIAL MANAGEMENT	3-0-0-4	3	3
G	B24CS2L03	DATA STRUCTURES LAB	0-0-3-2	3	2
H	B24CS2L04	NETWORKING LAB	0-0-3-2	3	2
I	B24MC2T04	UNIVERSAL HUMAN VALUE AND CONSTITUTIONAL RIGHTS	2-0-0-2	2	P/F
J	B24MC2T05	ENERGY CONSERVATION AND ENVIRONMENTAL SUSTAINABILITY	2-0-0-2	2	P/F
M	B24CSM3x	MINOR	3-1-0-5	4	4
TOTAL				32	22

SEMESTER 4

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24MA2T04C	GRAPH THEORY	3-1-0-5	4	4
B	B24CS2T04	OPERATING SYSTEMS	3-1-0-5	4	4
C	B24CS2T05	FORMAL LANGUAGES AND AUTOMATA THEORY	3-1-0-5	4	4
D	B24CS2T06	DATABASE MANAGEMENT SYSTEMS	3-1-0-4	4	3
E	B24HU2T03	ENTREPRENEURSHIP AND SOFTWARE MANAGEMENT SYSTEMS	2-1-0-4	3	3
F	B24CS2T07	SYSTEM SOFTWARE	2-1-0-4	3	3
G	B24CS2L05	LOGIC SYSTEM DESIGN AND OPERATING SYSTEMS LAB	0-0-3-2	3	2
H	B24CS2L06	DATABASE LAB	0-0-3-2	3	2
M	B24CSM4x	MINOR	3-1-0-5	4	4
N	B24CSH4x	HONOURS	3-1-0-5	4	4
TOTAL				36	25

SEMESTER 5

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24CS3T01	MICROPROCESSOR SYSTEMS	3-1-0-5	4	4
B	B24CS3T02	WEB PROGRAMMING	3-1-0-5	4	4
C	B24CS3T03	COMPILER DESIGN	3-1-0-5	4	4
D	B24CS3T04	ALGORITHM ANALYSIS AND DESIGN	3-1-0-5	4	4
E	B24CS3T05	ARTIFICIAL INTELLIGENCE	2-1-0-4	3	3
F	B24CS3P1x	PROGRAMME ELECTIVE I	2-1-0-4	3	3
G	B24CS3L07	SYSTEM SOFTWARE AND COMPILER DESIGN LAB	0-0-3-2	3	2
H	B24CS3L08	WEB PROGRAMMING LAB	0-0-3-2	3	2
M	B24CSM5x	MINOR	3-1-0-5	4	4
N	B24CSH5x	HONOURS	3-1-0-5	4	4
TOTAL				36	26

PROGRAMME ELECTIVE I

B24CS3P11	PROGRAMMING PARADIGMS (Bucket-I)
B24CS3P12	NUMBER THEORY (Bucket-II)
B24CS3P13	INTERNET OF THINGS (General-I)
B24CS3P14	DISTRIBUTED COMPUTING (General-II)

Note:- Six programme electives are offered from semester 5 onwards as per the curriculum. This curriculum envisages to offer a learner an opportunity to earn proficiency in one of two areas in Computer Science, namely Software Engineering (Bucket -I) and Security in Computing (Bucket -II). Six courses each from the above areas are included through Elective Buckets. Also learners have the option to choose general elective courses which are categorised as General-I and General-II instead of courses in buckets. Learners can choose any one of the buckets in semester 5 and they have to follow the same bucket in the subsequent semesters. Once they are out of the bucket by choosing General courses as electives, they will not be allowed to choose the subjects from any of the bucket. For example, a learner who is interested in the Software Engineering area may opt to take the elective courses -Programming Paradigms from Elective-I in S5, Object Modeling and Design from Elective-II in S6 and Software Performance and Scalability from Elective-III & Software Testing Architecture from Elective-IV in S7 and Business Analytics from Elective-V & Software Testing Platforms and Tools from Elective-VI in S8. The department may offer Elective Courses to enable students to utilize this opportunity, depending on the availability of faculty.

SEMESTER 6

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24CS3T06	SECURITY IN COMPUTING	3-1-0-5	4	4
B	B24CS3T07	DATA ANALYTICS SYSTEMS	3-1-0-5	4	4
C	B24CS3T08	CLOUD COMPUTING	3-1-0-5	4	4
D	B24CS3T09	COMPUTER GRAPHICS AND IMAGE PROCESSING	3-1-0-5	4	4
E	B24CS3P2x	PROGRAMME ELECTIVE II	2-1-0-4	3	3
F	B24CS3G1x	OPEN ELECTIVE I	2-1-0-4	3	3
G	B24CS3L09	MICROPROCESSOR LAB	0-0-3-2	3	2
H	B24CS3L10	MINI PROJECT	0-0-3-2	3	2
M	B24CSM6x	MINOR	3-1-0-5	4	4
N	B24CSH6x	HONOURS	3-1-0-5	4	4
TOTAL				36	26

PROGRAMME ELECTIVE II

B24CS3P21	OBJECT MODELING AND DESIGN (Bucket- I)
B24CS3P22A	CRYPTOGRAPHY AND NETWORK SECURITY(A) (Bucket- II)
B24CS3P23	LAN TECHNOLOGIES (General-I)
B24CS3P24	WEB MINING (General-II)

OPEN ELECTIVE I

B24CS3G11	INTRODUCTION TO DATA STRUCTURES
B24CS3G12	COMPUTER GRAPHICS
B24CS3G13	DIGITAL IMAGE PROCESSING

SEMESTER 7

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24CS4T01	BASICS OF MACHINE LEARNING	2-1-0-4	3	3
B	B24CS4P3x	PROGRAMME ELECTIVE III	2-1-0-4	3	3
C	B24CS4P4x	PROGRAMME ELECTIVE IV	2-1-0-4	3	3
D	B24CS4G2x	OPEN ELECTIVE II	2-1-0-4	3	3
E	B24HU4T04	DISASTER MANAGEMENT AND INDUSTRIAL SAFETY	2-1-0-4	3	3
G	B24CS4L11	COMPUTER GRAPHICS LAB	0-0-3-2	3	2
H	B24CS4L12	PROJECT PHASE I	0-0-6-2	6	3
J	B24CS4L13	SEMINAR	0-0-4-1	4	2
K	B24CS4T02	VIVA VOCE	0-0-0-0	-	1
M	B24CSM7x	MINOR	3-1-0-5	4	4
N	B24CSH7x	HONOURS	3-1-0-5	4	4
TOTAL				36	23

PROGRAMME ELECTIVE III

B24CS4P31	SOFTWARE PERFORMANCE AND SCALABILITY (Bucket -I)
B24CS4P32	INFORMATION AND DATA SECURITY (Bucket -II)
B24CS4P33	HIGH PERFORMACE COMPUTING (General -I)
B24CS4P34	MODERN DATABASES (General -II)

PROGRAMME ELECTIVE IV

B24CS4P41	SOFTWARE TESTING ARCHITECTURE (Bucket -I)
B24CS4P42	CYBER FORENSICS (Bucket -II)
B24CS4P43	EMBEDDED SYSTEMS (General -I)
B24CS4P44	CLIENT SERVER COMPUTING (General -II)

OPEN ELECTIVE II

B24CS4G21	INTRODUCTION TO DATABASES
B24CS4G22	BASICS OF LAN TECHNOLOGIES
B24CS4G23	INTRODUCTION TO MOBILE COMPUTING

SEMESTER 8

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A, B, C	B24MC400x	INTERNSHIP & MOOC COURSES (3 NUMBERS)	0-0-0-0	0	9
OR					
A	B24CS4P5x	PROGRAMME ELECTIVE V	2-1-0-4	3	3
B	B24CS4P6x	PROGRAMME ELECTIVE VI	2-1-0-4	3	3
C	B24CS4G3x	OPEN ELECTIVE III	2-1-0-4	3	3
AND					
H	B24CS4L14	PROJECT PHASE II	0-0-12-3	12	6
M	B24CSML1	MINOR PROJECT	0-0-3-12	3	6
N	B24CSHL1	HONOURS PROJECT	0-0-6-9	6	6
TOTAL				30	15

PROGRAMME ELECTIVE V

B24CS4P51	BUSINESS ANALYTICS (Bucket-I)
B24CS4P52	CLOUD SECURITY (Bucket-II)
B24CS4P53	BIOMETRICS (General- I)
B24CS4P54	PROGRAMMING IN R (General- II)

PROGRAMME ELECTIVE VI

B24CS4P61	SOFTWARE TESTING PLATFORMS AND TOOLS (Bucket-I)
B24CS4P62	BLOCKCHAIN TECHNOLOGIES (Bucket-II)
B24CS4P63	MOBILE COMPUTING (General- I)
B24CS4P64	SOFT COMPUTING (General-II)

OPEN ELECTIVE III

B24CS4G31	INTRODUCTION TO OPERATING SYSTEMS
B24CS4G32	BASICS OF CLIENT SERVER COMPUTING
B24CS4G33	INTRODUCTION TO DISTRIBUTED COMPUTING

PROGRAMME ELECTIVE LISTS

Elective	BUCKET-I (Software Engineering)	BUCKET-II (Security in Computing)	General-I	General-II
I	PROGRAMMING PARADIGMS	NUMBER THEORY	INTERNET OF THINGS	DISTRIBUTED COMPUTING
II	OBJECT MODELING AND DESIGN	CRYPTOGRAPHY AND NETWORK SECURITY	LAN TECHNOLOGIES	WEB MINING
III	SOFTWARE PERFORMANCE AND SCALABILITY	INFORMATION AND DATA SECURITY	HIGH PERFORMANCE COMPUTING	MODERN DATABASES
IV	SOFTWARE TESTING ARCHITECTURE	CYBER FORENSICS	EMBEDDED SYSTEMS	CLIENT SERVER COMPUTING
V	BUSINESS ANALYTICS	CLOUD SECURITY	BIOMETRICS	PROGRAMMING IN R
VI	SOFTWARE TESTING PLATFORMS AND TOOLS	BLOCK CHAIN TECHNOLOGIES	MOBILE COMPUTING	SOFT COMPUTING

MINOR

SEMESTER	BUCKET – I (Programming Methodologies)	BUCKET – II (Machine Learning)	BUCKET – III (Networking)
III	PYTHON PROGRAMMING	MATHEMATICS FOR MACHINE LEARNING	DATA COMMUNICATION CONCEPTS
IV	OBJECT ORIENTED PROGRAMMING IN JAVA	INTRODUCTION TO MACHINE LEARNING	INTRODUCTION TO COMPUTER NETWORKS
V	DATABASE PROGRAMMING	CONCEPTS IN DEEP LEARNING	CLIENT SERVER SYSTEMS
VI	WEB TECHNOLOGIES	PYTHON FOR MACHINE LEARNING	WIRELESS NETWORKS AND IOT APPLICATIONS
VII	MINOR PROJECT	MINOR PROJECT	MINOR PROJECT
VIII	MINOR PROJECT	MINOR PROJECT	MINOR PROJECT

Note:-Learners can choose any one of the buckets in semester 3 and they have to follow the same bucket in the subsequent semesters.

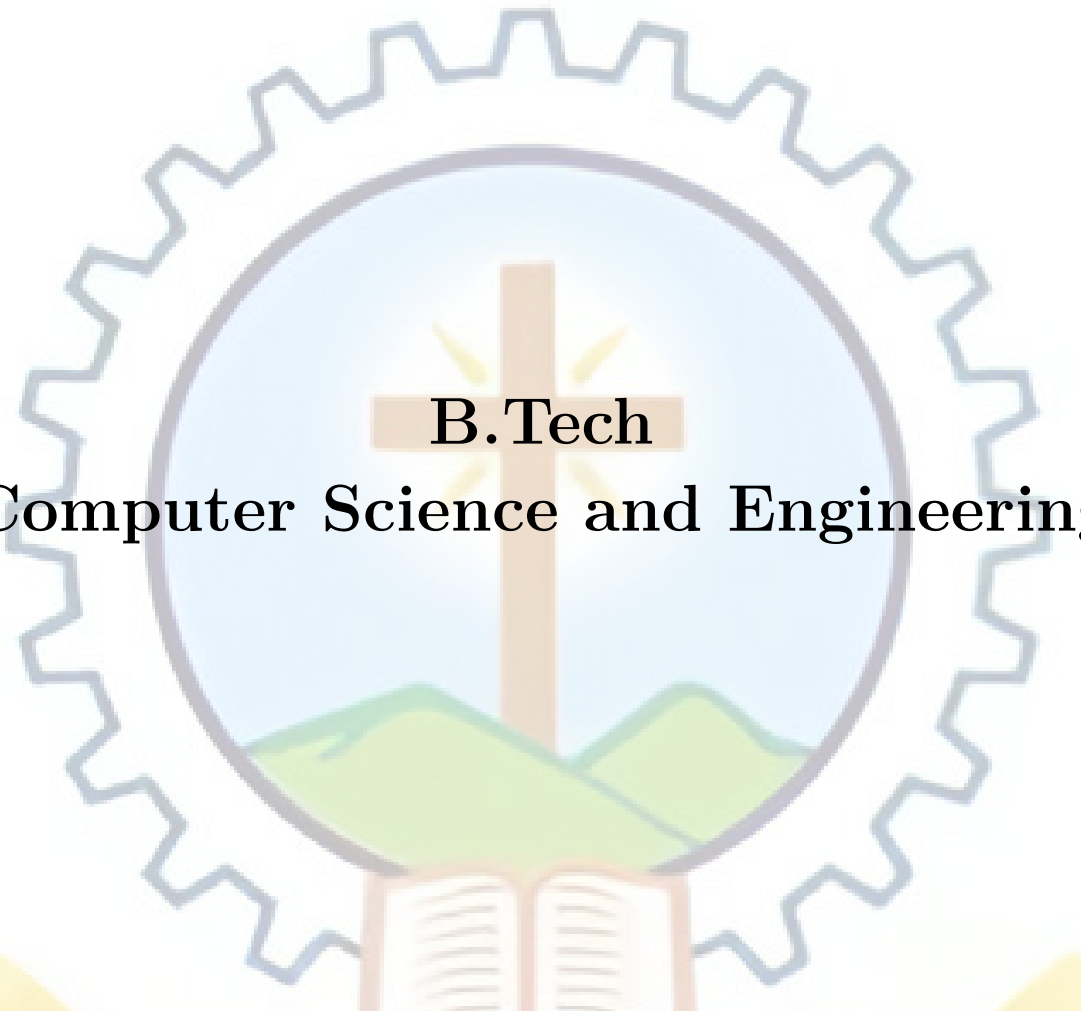
HONOURS

SEMESTER	BUCKET- I (Computer Vision)	BUCKET- II (Computational Biology)	BUCKET- III (Formal Methods)
IV	ADVANCED TOPICS IN COMPUTER GRAPHICS	COMPUTATIONAL FUNDAMENTALS FOR BIOINFORMATICS	PRINCIPLES OF PROGRAM ANALYSIS AND VERIFICATION
V	ADVANCED CONCEPTS IN COMPUTER VISION	MACHINE LEARNING IN COMPUTATIONAL BIOLOGY	PRINCIPLES OF MODEL CHECKING
VI	IMAGE AND VIDEO PROCESSING	COMPUTATIONAL HEALTH INFORMATICS	THEORY OF COMPUTABILITY AND COMPLEXITY
VII	SURVEILLANCE VIDEO ANALYTICS	COMPUTER AIDED DRUG DESIGN	LOGIC FOR COMPUTER SCIENCE
VIII	HONOURS PROJECT	HONOURS PROJECT	HONOURS PROJECT

Note:- Learners can choose any one of the buckets in semester 4 and they have to follow the same bucket in the subsequent semesters.

MAR ATHANASIOUS COLLEGE OF ENGINEERING

Government Aided, Autonomous Institution
Kothamangalam, Kerala, India

The logo features a large blue gear-like outer circle. Inside is a light blue circle containing a brown cross with yellow rays emanating from its center. Below the cross are green hills. At the bottom of the inner circle is an open book with white pages. The text 'B.Tech' is written in black on a brown rectangular background across the cross, and 'Computer Science and Engineering' is written in black below it.

B.Tech **Computer Science and Engineering**

SEMESTER 1

SYLLABUS

KNOWLEDGE IS POWER

SEMESTER 1

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24MA1T01	LINEAR ALGEBRA AND MULTIVARIABLE CALCULUS	3-1-0-5	4	4
B	B24ES1T01A	PROBLEM SOLVING & PROGRAMMING TECHNIQUES (A)	2-1-0-4	3	3
C	B24PH1T01A	ENGINEERING PHYSICS (A)	2-1-0-4	3	3
D	B24CY1T01A	ENGINEERING CHEMISTRY (A)	2-1-0-4	3	3
E	B24ES1T02	BASICS OF ELECTRICAL & ELECTRONICS ENGINEERING	2-2-0-5	4	4
F	B24ES1L02	BASIC ELECTRICAL AND ELECTRONICS WORKSHOP	0-0-2-1	2	1
G	B24ES1L01A	PROGRAMMING LAB (A)	0-0-3-2	3	2
H	B24PC1L01A	PHYSICS(A) & CHEMISTRY (A) LAB	0-0-2-2	2	1
I	B24MC1T01	LIFE SKILLS	1-0-1-2	2	P/F
J	B24MC1T02	DESIGN THINKING	1-1-0-2	2	P/F
K	B24MC1L01	YOGA AND SPORTS	0-1-1-1	2	P/F
TOTAL				30	21

B24MA1T01	LINEAR ALGEBRA AND MULTIVARIABLE CALCULUS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		

Preamble

This course introduces students to some basic mathematical ideas and tools which are at the core of any engineering course. A brief course in Linear Algebra familiarises students with some basic techniques in matrix theory which are essential for analyzing linear systems. The calculus of functions of one or more variables taught in this course are useful in modelling and analyzing physical phenomena involving continuous change of variables or parameters and have applications across all branches of engineering.

Prerequisites: Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Solve systems of linear equations using matrix methods and determine eigenvalues, eigenvectors and canonical forms of quadratic forms.(Cognitive Knowledge level: Apply)
CO 2	Calculate partial and total derivatives of multivariable functions and determine their extrema using differential calculus techniques.(Cognitive Knowledge level: Apply)
CO 3	Apply double and triple integrals to determine areas, volumes, mass and centre of gravity of geometrical regions.(Cognitive Knowledge level: Apply)
CO 4	Calculate derivatives and line integrals of vector functions and apply them to solve problems in vector calculus.(Cognitive Knowledge level: Apply)
CO 5	Apply Green's, Stokes' and Divergence theorems to calculate line, surface and flux integrals and model engineering field problems.(Cognitive Knowledge level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1			1						
CO 2	2	1				1					
CO 3	2	1									
CO 4	2	1									
CO 5	2	1		1	1	1	1				1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (Linear Algebra)

(Text 2: Relevant topics from 7.3, 7.4, 7.5, 8.1,8.3,8.4)

Systems of linear equations, Solution by Gauss elimination, row echelon form and rank of a matrix, fundamental theorem for linear systems (homogeneous and non-homogeneous, without proof), Eigen values and eigenvectors. Diagonalization of matrices, orthogonal transformation, quadratic forms and their canonical forms.

MODULE 2 (Multivariable Calculus-Differentiation)

(Text 1: Relevant topics from sections 13.3, 13.4, 13.5, 13.8)

Partial derivatives, partial derivatives of functions of more than two variables, higher order partial derivatives, differentials and local linearity, The chain rule, Maxima and Minima of functions of two variables, extreme value theorem (without proof), relative extrema.

MODULE 3 ((Multivariable Calculus-Integration))

(Text 1: Relevant topics from sections 14.1, 14.2, 14.3, 14.5, 14.6, 14.8)

Double integrals (Cartesian), reversing the order of integration, Change of coordinates (Cartesian to polar), finding areas using double integrals, mass and centre of gravity of inhomogeneous laminas using double integral. Triple integrals, volume calculated as triple integral, triple integral in cylindrical and spherical coordinates (computations involving spheres, cylinders).

MODULE 4 (Calculus of vector functions)

(Text 1: Relevant topics from sections 12.1, 12.2, 12.6, 13.6, 15.1, 15.2, 15.3)

Vector valued function of single variable, derivative of vector function and geometrical interpretation, motion along a curve-velocity, speed and acceleration. Concept of scalar and vector fields, Gradient and its properties, directional derivative, divergence and curl, Line integrals of vector fields, work as line integral, Conservative vector fields, independence of path and potential function(results without proof).

MODULE 5 (Vector integral theorems)

(Text 1: Relevant topics from sections 15.4, 15.5, 15.6, 15.7, 15.8)

Green's theorem (for simply connected domains, without proof) and applications to evaluating line integrals and finding areas. Surface integrals over surfaces of the form $z = g(x, y)$, $y = g(x, z)$ or $x = g(y, z)$, Flux integrals over surfaces of the form $z = g(x, y)$, $y = g(x, z)$ or $x = g(y, z)$, divergence theorem (without proof) and its applications to finding

flux integrals, Stokes' theorem (without proof) and its applications to finding line integrals of vector fields and work done.

Text Books

1. H. Anton, I. Biven, S. Davis, "Calculus", Wiley, 10th edition, 2015.
2. Erwin Kreyszig, Advanced Engineering Mathematics, 10th edition, John Wiley & Sons, 2015.

Reference Books

4. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, 9th Edition, Pearson, Reprint, 2002.
5. J. Stewart, Essential Calculus, Cengage, 2nd edition, 2017.
6. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 43rd Edition, 2015.
7. Peter O Neil, Advanced Engineering Mathematics, 7th Edition, Thomson, 2007.
8. Veerarajan T. Engineering Mathematics for first year", Tata McGraw - Hill, 2008.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 hours
Module 1: Linear Algebra		9 hours
1.1	Systems of linear equations, Solution by Gauss elimination	1
1.2	Row echelon form, finding rank from row echelon form, fundamental theorem for linear systems	2
1.3	Eigen values and eigen vectors	2
1.4	Diagonalization of matrices	2
1.5	Orthogonal transformation, quadratic forms and their canonical forms.	2
Module 2: Multivariable Calculus - Differentiation		9 hours
2.1	Partial derivatives	2
2.2	Differentials, Local Linear approximations	2
2.3	Chain rule, total derivative	2

2.4	Maxima and minima	3
Module 3: Multivariable Calculus - Integration		9 hours
3.1	Double integrals (Cartesian)-evaluation	2
3.2	Change of order of integration in double integrals, change of coordinates (Cartesian to polar)	2
3.3	Finding areas, mass and centre of gravity of plane laminas	2
3.4	Triple integrals, volume calculated as triple integral, triple integral in cylindrical and spherical coordinates.	3
Module 4: Calculus of Vector Functions		9 hours
4.1	Vector valued function of a scalar variable - derivative of vector valued function of scalar variable t-geometrical meaning	2
4.2	Motion along a curve-speed, velocity, acceleration	1
4.3	Gradient and its properties, directional derivative, divergence and curl	3
4.4	Line integrals with respect to arc length, line integrals of vector fields. Work done as line integral	2
4.5	Conservative vector field, independence of path, potential function	1
Module 5: Vector Integral Theorems		9 hours
5.1	Green's theorem and it's applications	2
5.2	Surface integrals, flux integral and their evaluation	3
5.3	Divergence theorem and applications	2
5.4	Stokes theorem and applications	2

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1): Solve systems of linear equations, diagonalize matrices and characterise quadratic forms.

1. A is a real matrix of order 3×3 and $X = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$. What can you say about the solution of $AX = 0$ if rank of A is 2 ? 3 ?

2. Given $A = \begin{bmatrix} 3 & 0 & 2 \\ 0 & 2 & 0 \\ -2 & 0 & 0 \end{bmatrix}$, find an orthogonal matrix P that diagonalizes A .

3. The matrix $A = \begin{bmatrix} -2 & 2 & -3 \\ 2 & 1 & -6 \\ -1 & -2 & 0 \end{bmatrix}$ has an eigenvalue 5 with corresponding eigenvector

$X = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix}$. Find $A^5 X$.

Course Outcome 2 (CO 2): Compute the partial and total derivatives and maxima and minima of multivariable functions.

1. Find the slope of the surface $z = x^2y + 5y^3$ in the x -direction at the point $(1, -2)$.
2. Given the function $w = xy + z$, use the chain rule to find the instantaneous rate of change of w at each point along the curve $x = \cos t$, $y = \sin t$, $z = t$.
3. Determine the dimensions of a rectangular box open at the top, having a volume 32 cubic ft and requiring the least amount of material for its construction.

Course Outcome 3 (CO 3): Compute multiple integrals and apply them to find areas and volumes of geometrical shapes, mass and centre of gravity of plane laminas.

1. Evaluate $\iint_D (x + 2y) dA$ where D is the region bounded by the parabolas $y = 2x^2$ and $y = 1 + x^2$.
2. Explain how you would find the volume under the surface $z = f(x, y)$ and over a specific region D in the xy plane using triple integral?
3. Find the mass and centre of gravity of a triangular lamina with vertices $(0,0)$, $(2,1)$, $(0,3)$ if the density function is $f(x, y) = x + y$.

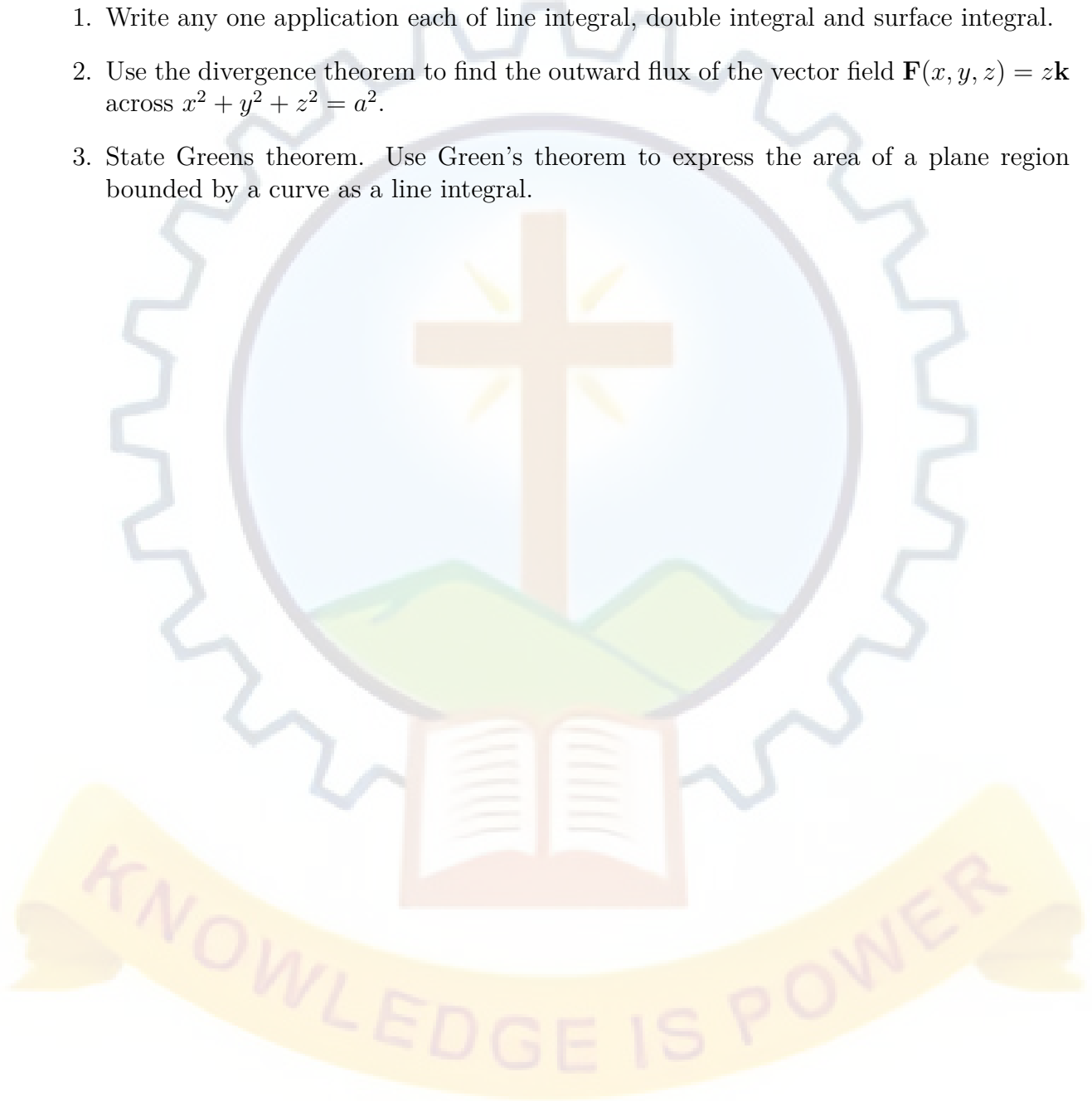
Course Outcome 4 (CO 4): Compute the derivatives and line integrals of vector functions and learn their applications

1. How would you calculate the speed, velocity and acceleration at any instant of a particle moving in space whose position vector at time t is $\mathbf{r}(t)$?
2. Find the work done by the force field $\mathbf{F} = (e^x - y^3) \mathbf{i} + (\cos y + x^3) \mathbf{j}$ on a particle that travels once around the unit circle centered at origin having radius 1.

3. When do you say that a vector field is conservative? What are the implications if a vector field is conservative?

Course Outcome 5 (CO 5): Evaluate surface and volume integrals and learn their inter-relations and applications

1. Write any one application each of line integral, double integral and surface integral.
2. Use the divergence theorem to find the outward flux of the vector field $\mathbf{F}(x, y, z) = z\mathbf{k}$ across $x^2 + y^2 + z^2 = a^2$.
3. State Greens theorem. Use Green's theorem to express the area of a plane region bounded by a curve as a line integral.



MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM

FIRST SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2024

Course Code: B24MA1T01

Course Name: **LINEAR ALGEBRA AND MULTIVARIABLE CALCULUS**

Common to all branches

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Determine the rank of the matrix $\begin{bmatrix} 1 & 2 & -1 \\ -2 & -4 & 2 \\ 3 & 6 & -3 \end{bmatrix}$

2. Write down the eigen values of $A = \begin{bmatrix} 2 & 0 \\ 0 & -1 \end{bmatrix}$

3. Find $f_x(1,3)$ and $f_y(1,3)$ for the function $f(x,y) = 2x^3y^2 + 2y + 4x$.

4. Show that the function $u(x,t) = \sin(x - ct)$ is a solution of the equation $\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2}$.

5. Use double integral to find the area of the region enclosed between the parabola $y = \frac{x^2}{2}$ and the line $y = 2x$.

6. Use polar coordinates to evaluate the area of the region bounded by $x^2 + y^2 = 4$, the line $y = x$ and the y axis in the first quadrant.

7. Is the vector $\mathbf{r}$ where $\mathbf{r} = x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$ conservative. Justify your answer.

8. Find a unit vector normal to the surface $x^3 + y^3 + 3xyz = 3$ at the point $(1,2,-1)$.
9. What is the outward flux of $\mathbf{F}(x, y, z) = x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$ across any unit cube.
10. What is the relationship between Green's theorem and Stokes theorem?

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Solve the following system of equations.

$$y + z - 2w = 0$$

$$2x - 3y - 3z + 6w = 2$$

$$4x + y + z - 2w = 4$$

7

- (b) Find the eigen values and eigen vectors of the matrix.

$$\begin{bmatrix} 2 & 2 & -3 \\ 2 & 1 & -6 \\ -1 & 2 & 0 \end{bmatrix}$$

7

OR

12. (a) Diagonalize the matrix

$$\begin{bmatrix} -1 & 2 & -2 \\ 2 & 4 & 1 \\ 2 & 4 & 1 \end{bmatrix}.$$

7

- (b) What kind of conic section the quadratic form $3x^2 + 22xy + 3y^2 = 0$ represents? Transform it to principal axes.

7

13. (a) Find the local linear approximation to $f(x, y) = \sqrt{x^2 + y^2}$ at the point $(3,4)$. Use it to approximate $f(3.04, 3.98)$.

7

- (b) Let $w = \sqrt{x^2 + y^2 + z^2}$, $x = \cos\theta$, $y = \sin\theta$, $z = \tan\theta$. Use chain rule to find $\frac{dw}{d\theta}$ when $\theta = \frac{\pi}{4}$.

7

OR

14. (a) Let $z = f(x, y)$ where $x = r\cos\theta$, $y = r\sin\theta$, prove that

$$\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2 = \left(\frac{\partial z}{\partial r}\right)^2 + \frac{1}{r^2} \left(\frac{\partial z}{\partial \theta}\right)^2$$

7

- (b) Locate all relative maxima, relative minima and saddle points of $f(x, y) = xy + \frac{a^3}{x} + \frac{b^3}{y}$, $(a \neq 0, b \neq 0)$.

7

15. (a) Evaluate $\iint_D (2x^2y + 9y^3) dx dy$ where D is the region bounded by $y = \frac{2}{3}x$ and $y = 2\sqrt{x}$.

- (b) Evaluate $\int_0^4 \int_{\sqrt{y}}^2 e^{x^3} dx dy$ by changing the order of integration.

7

OR

16. (a) Find the volume of the solid bounded by the cylinder $x^2 + y^2 = 4$ and the planes $y + z = 4$ and $z = 0$. 7
- (b) Evaluate $\iiint \sqrt{1 - x^2 - y^2 - z^2} dx dy dz$, taken throughout the volume of the sphere $x^2 + y^2 + z^2 = 1$. 7
17. (a) Prove that the force field $\mathbf{F} = e^y \mathbf{i} + xe^y \mathbf{j}$ is conservative in the entire xy-plane. 7
- (b) Find the work done in moving a particle along a straight line from (0,0,0) to (2,1,3) by the force $\mathbf{F} = 3x^2 \mathbf{i} + (2xz - y) \mathbf{j} + z \mathbf{k}$. 7

OR

18. (a) Find the divergence of the vector field $\mathbf{F} = x^3 y^2 z \mathbf{i} + xyz^3 \mathbf{j} + xyz^2 \mathbf{k}$ at (1,1,1). 7
- (b) Find the work done by the force field $\mathbf{F}(x, y, z) = xy \mathbf{i} + yz \mathbf{j} + xz \mathbf{k}$ along C where C is the curve $\mathbf{r}(t) = t \mathbf{i} + t^2 \mathbf{j} + t^3 \mathbf{k}$. 7
19. (a) Use divergence theorem to find the outward flux of the vector field $\mathbf{F} = 2x \mathbf{i} + 3y \mathbf{j} + z^3 \mathbf{k}$ across the unit cube bounded by $x = 0, y = 0, z = 0, x = 1, y = 1, z = 1$. 7
- (b) Find the circulation of $\mathbf{F} = (x - z) \mathbf{i} + (y - x) \mathbf{j} + (z - xy) \mathbf{k}$ using Stokes theorem around the triangle with vertices A(1,0,0), B(0,2,0) and C(0,0,1). 7

OR

20. (a) Use divergence theorem to find the volume of the cylindrical solid bounded by $x^2 + 4x + y^2 = 7, z = -1, z = 4$ given the vector field $\mathbf{F} = x \mathbf{i} + y \mathbf{j} + z \mathbf{k}$ across surface of the cylinder. 7
- (b) Use Stokes theorem to evaluate $\int_C \mathbf{F} \cdot d\mathbf{r}$ where $\mathbf{F} = x^2 \mathbf{i} + 3x \mathbf{j} - y^3 \mathbf{k}$ where C is the circle $x^2 + y^2 = 1$ in the xy-plane with counterclockwise orientation looking down the positive z-axis. 7

B24ES1T01A	PROBLEM SOLVING AND PROGRAMMING TECHNIQUES (A)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4		2024

Preamble

This course shall prepare Engineering Graduates to write versatile C programs for solving computational problems that they come across in their professional life. The subject covers the basics of C programming, array handling, string manipulations, function creation, structure and pointer operations and file processing. On completing this course a learner will be able to write efficient C programs to solve real world computational problems.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Analyze a computational problem and try to solve it using algorithms, flowcharts and also develop C programs from them using Arithmetic, Logical, Relational and Bitwise operators. (Cognitive Knowledge Level: Understand)
CO 2	Develop C programs with branching and looping statements for processing arrays and matrices (Cognitive Knowledge Level: Apply)
CO 3	Divide a given computational problem into a number of modules and develop functions to find the solution to the computational problem and also create programs for string processing (Cognitive Knowledge Level: Apply)
CO 4	Develop C programs which use structures and pointers for data processing and parameter passing (Cognitive Knowledge Level: Apply)
CO 5	Develop C programs for file processing (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	3	3		1	1	1	1	1	1
CO2	3	3	3	3		1	1	1		1	1
CO3	3	3	3	3		1	1	1		1	1
CO4	3	3	3	3		1	1	1		1	1
CO5	3	3	3	3		1	1	1		1	1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember			
Understand	40	40	40
Apply	60	60	60
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Problem solving:

Problem solving using Algorithms, Pseudocode and Flowcharts.

C fundamentals:

Character set, Constants, Identifiers, Keywords, Basic data types, Variables, Operators and its precedence, Bitwise operators, Expressions, Statements, Input and Output statements– Structure of a C program– simple programs.

MODULE 2 (9 hours)

Control statements:

If, if-else, nested if, switch, while, do-while, for, break & continue, nested loops. Single-dimensional arrays – defining an array, array initialization, accessing array elements, Enumerated data type, Two-dimensional arrays – defining a two-dimensional array – Programs for matrix processing - Programs for Sequential search, Bubble sort.

MODULE 3 (8 hours)

Strings:

Declaring a string variable, reading and displaying strings, string-related library functions – Programs for string matching.

Functions:

Function definition, function call, function prototype, parameter passing – Recursion – Passing array to function. Macros: Defining and calling macros.

MODULE 4 (8 hours)

Structures:

Defining a structure variable, accessing members, array of structures, passing structure to function. Union, Pointers: declaration, operations on pointers, passing pointer to a function, accessing array elements using pointers, processing strings using pointers, pointer to pointer, array of pointers, pointer to function, pointer to structure, Dynamic memory allocation.

MODULE 5 (5 hours)

Files:

Different types of files in C – Opening & Closing a file – Writing to and Reading from a file – Processing files – Library functions related to file – fseek(), ftell(), fread(), fwrite().
Storage Class associated with variables: automatic, static, external and register.

Text Books

1. Programming with C - Byron S. Gottfried, Tata McGraw Hill
2. Computer Programming in C - Kerninghan & Ritchie, PHI

Reference Books

1. Programming in C - Stephen C. Kochan, CBS publishers.
2. Programming in C – E. Balaguruswamy , Mc Graw Hill.
3. Let us C – Yashwant Kanetkar, BPB Publications.
4. A Book on C – Al Kelley and Ira Pohl, Addison-Wesley.
5. Mastering Turbo C - Stan Kelly Bootle, BPB Publications.
6. Pointers in C - Yashwant Kanetkar, BPB Publications.
7. The Spirit of C- by Munish cooper, Jaico Books.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: C fundamentals		6 hours
1.1	Problem solving using Algorithms, Pseudocode and Flowcharts.	1
1.2	C fundamentals: Character set, Constants, Identifiers.	1

1.3	Keywords, Basic data types, Variables.	1
1.4	Operators and its precedence, bitwise operators.	1
1.5	Expressions, Statements, Input and Output statements.	1
1.6	Structure of a C program– simple programs.	1
Module 2: Control statements		9 hours
2.1	Control statements: if, if-else, nested if .	1
2.2	Switch, while loop, do-while loop.	1
2.3	For loop, break & continue statements, nested loops.	1
2.4	Single dimensional arrays – defining an array, array initialization, accessing array elements.	1
2.5	Two-dimensional arrays – defining a two-dimensional array.	1
2.6	Programs for matrix processing.	1
2.7	Programs for sequential search.	1
2.8	Bubble sort.	1
2.9	Enumerated data type.	1
Module 3: Strings and Functions		8 hours
3.1	Strings: declaring a string variable, reading and displaying strings.	1
3.2	String related library functions.	1
3.3	Programs for string matching.	1
3.4	Functions: Function definition, Function call.	1
3.5	Function prototype, Parameter passing.	1
3.6	Recursion.	1
3.7	Passing array to function.	1
3.8	Macros: Defining and calling macros.	1
Module 4: Structures		8 hours
4.1	Structures: defining a structure variable, accessing members.	1
4.2	Array of structures, passing structure to function.	1
4.3	Union.	1
4.4	Pointers: declaration, operations on pointers, pointer to a function.	1

4.5	Accessing array elements using pointers, Processing strings using pointers.	1
4.6	Pointer to pointer, Array of pointers.	1
4.7	Pointer to function, Pointer to structure.	1
4.8	Dynamic memory allocation.	1
Module 5: Files		5 hours
5.1	Different types of files in C, Opening & Closing a file.	1
5.2	Writing to and Reading from a file, Processing file.	1
5.3	Library functions related to file – fseek(), ftell(), fread(), fwrite().	2
5.4	Storage Class associated with variables: automatic, static, external and register.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. Write an algorithm and pseudocode to check if a given number is an Armstrong number or not.
2. Draw a flow chart to check if a given number is an Armstrong number or not.

Course Outcome 2 (CO 2):

1. Write a C program to find the smallest number, largest number and the number of occurrences from a set of numbers.
2. Write a C program to add two matrices.

Course Outcome 3 (CO 3):

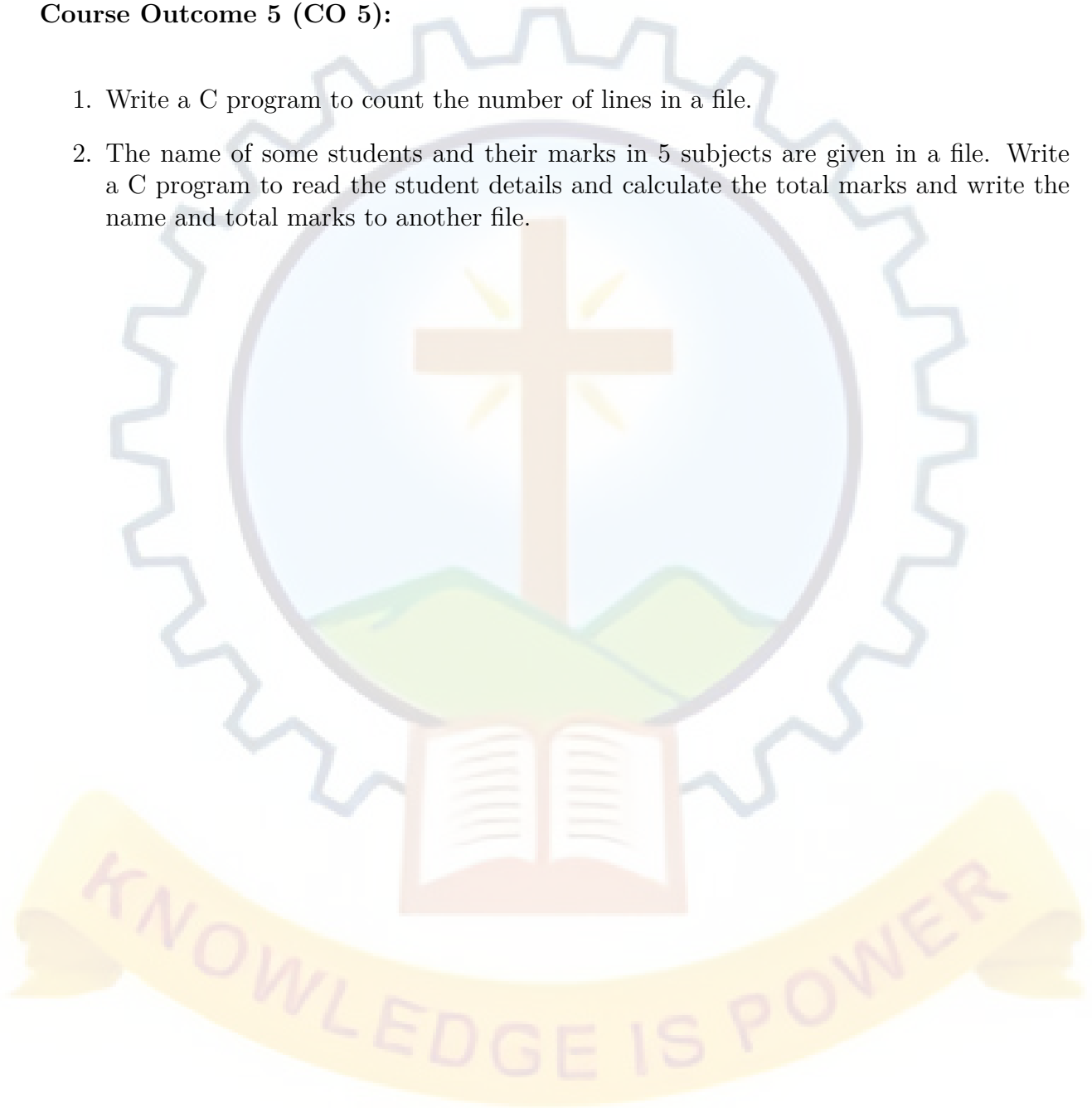
1. Write a C program to find whether a string is present in another string.
2. Write functions to accept an N X N matrix and find the row sum and column sum of the matrix.

Course Outcome 4 (CO 4):

1. Write a C program to find the difference between two time intervals using structure.
2. Write a C program to check if a given string is a palindrome using pointers.

Course Outcome 5 (CO 5):

1. Write a C program to count the number of lines in a file.
2. The name of some students and their marks in 5 subjects are given in a file. Write a C program to read the student details and calculate the total marks and write the name and total marks to another file.



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

First SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2024

Course Code: B24ES1T01A

***Course Name: PROBLEM SOLVING AND PROGRAMMING
TECHNIQUES (A)***

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Draw a flow chart to find the largest of three numbers.
2. Write a C program to convert Fahrenheit temperature to Celsius.
3. Differentiate between while loop and do-while loop.
4. Write a C program to find all the factors of a number.
5. Explain any 3 string handling functions using examples.
6. Differentiate between macros and functions.
7. What are the advantages of using structure in the C language?
8. Explain pointer to a pointer with an example.
9. Write any three file handling functions in C.
10. What is a static variable? When should it be used?

PART B

Answer any one question from each module. Each question carries 14 marks.

11. Explain linear search with an example. Draw a flowchart and write pseudo code to perform linear search on an array of numbers. 14

OR

12. (a) Write a C program to find the area of a triangle given the length of three sides of the triangle. 7
(b) Write a C program to find the Area and Circumference of a Circle given the radius of the circle. 7
13. (a) Write a C program to find the transpose of a matrix. 7
(b) Write a C program to sort an array of numbers using bubble sort. 7

OR

14. (a) Write a C program to find the sum of the first and last digit of a number. 7
(b) Write a C program to print all the prime numbers between 100 to 200. 7
15. (a) Explain any 4 string handling functions in C programming. 7
(b) Write a C program to reverse a string without using string handling functions. 7

OR

16. (a) What is the purpose of function declaration and function definition and function call? With examples illustrate their syntax. 7
(b) What is recursion? Write a C program to display the Fibonacci series using recursive function. 7
17. (a) Write a C program to: 7
i. Create a structure with fields: Name, Address, Date of Birth.
ii. Read the above details for five students from user and display the details.
(b) Differentiate between an array of pointers and a pointer to an array. 7

OR

18. (a) What are the different dynamic memory allocation functions available in the C language? 7
(b) Write a C program to reverse a string using pointers. 7
19. (a) What are different storage classes in C? Give examples for each. 7
(b) Explain any 5 file handling functions in C? 7

OR

20. (a) Write a C program to count the number of lines in a text file. 7
(b) Write a C program to read a text file and replace all vowels in the text file with the character 'x' and write it to another file. 7

B24PH1T01A	ENGINEERING PHYSICS (A)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

The aim of this course is to equip students with a solid foundation in physics principles and knowledge of their engineering applications. This will enhance the students' ability to analyze and solve complex engineering problems. Ultimately, the goal is to produce graduates who are well prepared to tackle real-world engineering challenges with a deep understanding of the underlying physical principles.

Prerequisites

Higher secondary level Physics and Mathematics

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the principles of lasers and optical fibers, including Einstein's relations and numerical aperture, and apply them to understand intensity-modulated sensors.(Cognitive Knowledge Level: Apply)
CO 2	Formulate and solve the time-independent Schrödinger equation for a particle in a one-dimensional box, interpreting wave functions and energy eigenvalues.(Cognitive Knowledge Level: Apply)
CO 3	Analyze electrical conduction in solids using Fermi-Dirac statistics and evaluate the charge carrier concentrations in intrinsic semiconductors.(Cognitive Knowledge Level: Apply)
CO 4	Evaluate carrier concentrations in extrinsic semiconductors and interpret the variation of the Fermi level with temperature and doping concentration.(Cognitive Knowledge Level: Apply)
CO 5	Illustrate the formation, energy band diagrams, and working of PN junctions and photonic devices (LEDs, photodiodes, solar cells) for sustainable energy and optoelectronic applications.(Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO 1	2	1		1	1	1			1		1
CO 2	2	1							1		1
CO 3	2	1							1		
CO 4	2	1							1		
CO 5	2	1			1				1		1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	50	50	50
Apply	20	20	20
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Laser & Fibre Optics:

Optical processes - Absorption, Spontaneous emission and stimulated emission, Einstein's relations. Principle of laser - conditions for sustained lasing - components of laser - Population inversion - energy source - Pumping, Metastable states - active medium, optical resonator. Construction and working of Ruby laser. Optic fiber-Principle of propagation of light, Numerical aperture – Derivation. Applications of fibers - Intensity modulated sensors.

MODULE 2 (7 hours)

Quantum Mechanics:

Introduction - Concept of uncertainty and conjugate observables (qualitative), Uncertainty principle (statement only), Wave function, its properties and physical interpretation, Formulation of time dependent and time independent Schrodinger equations, Particle in a one dimensional box - Derivation of energy eigenvalues and normalized wave function.

MODULE 3 (8 hours)

Semiconductor Physics I :

Electrical Conduction in solids - Density of states function (no derivation), the Fermi-Dirac Probability function, Fermi energy and its physical significance, Charge carriers in semiconductors - Equilibrium distribution of electrons and holes, the n_0 and p_0 equations, Intrinsic carrier concentration n_i , Intrinsic Fermi level position and its dependence on temperature.

MODULE 4 (8 hours)

Semiconductor Physics II :

Extrinsic semiconductors - P type semiconductor, N type semiconductor, Carrier concentration in N type semiconductor, Variation of fermi level with temperature, Variation of fermi level with donor concentration, Carrier concentration in P type semiconductor, Variation of fermi level with temperature, Variation of fermi level with acceptor concentration.

MODULE 5 (6 hours)

Semiconductor Devices:

Formation of PN junction, Energy band diagram of PN junction - Qualitative description of charge flow across a PN junction - Forward and reverse biased PN Junctions, Photonic devices (Qualitative treatment only) - Light Emitting Diode, Photo detectors (Junction and PIN photodiodes), Solar cells.

Text Books

1. Aruldas G., "Engineering Physics", PHI Pvt. Ltd., 2015.
2. M.N. Avadhanulu, P.G. Kshirsagar, TVS Arun Murthy, "A Textbook of Engineering Physics", S.Chand & Co., Revised Edition, 2019.
3. Donald A. Neamen, "Semiconductor Physics and Devices - Basic Principles", McGraw Hill, Fourth Edition, 2012.

Reference Books

1. Arthur Beiser, "Concepts of Modern Physics ", Tata McGraw Hill Publications, 6th Edition 2003.
2. D.K. Bhattacharya, Poonam Tandon, "Engineering Physics", Oxford University Press, 2015.
3. Md.N.Khan & S.Panigrahi "Principles of Engineering Physics 1&2", Cambridge University Press, 2016.
4. Aruldas G., "Engineering Physics", PHI Pvt. Ltd., 2015.
5. S.M. Sze, "Physics of Semiconductor Devices", John Wiley & Sons, 1969.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Laser and Fibre Optics		7 hours
1.1	Optical processes - Absorption, Spontaneous emission and stimulated emission, Einstein's relations	2

1.2	Principle of laser - conditions for sustained lasing - components of laser - Population inversion - energy source - Pumping, Metastable states - active medium, optical resonator.	2
1.3	Construction and working of Ruby laser.	1
1.4	Optic fiber-Principle of propagation of light, Numerical aperture – Derivation	1
1.5	Applications of fibers - Intensity modulated sensors .	1
Module 2: Quantum Mechanics		7 hours
2.1	Introduction - Concept of uncertainty and conjugate observables (qualitative), Uncertainty principle (statement only),	1
2.2	Wave function, its properties and physical interpretation.	1
2.3	Formulation of time dependent and time independent Schrodinger equations, Particle in a one dimensional box - Derivation of energy eigenvalues and normalized wave function, Numerical Problems.	5
Module 3: Semiconductor Physics I		8 hours
3.1	Electrical Conduction in solids - Density of states function (no derivation), the Fermi-Dirac Probability function, Fermi energy and its physical significance	2
3.2	Charge carriers in semiconductors - Equilibrium distribution of electrons and holes, the n_0 and p_0 equations.	3
3.3	Intrinsic carrier concentration n_i Intrinsic Fermi level position and its dependence on temperature.	3
Module 4: Semiconductor Physics II		8 hours
4.1	Extrinsic semiconductors - P type semiconductor, N type semiconductor.	2
4.2	Carrier concentration in N type semiconductor, Variation of fermi level with temperature, Variation of fermi level with donor concentration	3
4.3	Carrier concentration in P type semiconductor, Variation of fermi level with temperature, Variation of fermi level with acceptor concentration	3
Module 5: Semiconductor Devices		6 hours

5.1	Formation of PN junction, Energy band diagram of PN junction - Qualitative description of charge flow across a PN junction - Forward and reverse biased PN Junctions, the ideal diode equation (no derivation).	3
5.2	Photonic devices (Qualitative treatment only) - Light Emitting Diode, Photo detectors (Junction and PIN photodiodes), Solar cells.	3

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. Describe the principle of LASER.
2. Why are metastable levels needed in a LASER?
3. Write a note on intensity modulated sensors.

Course Outcome 2 (CO 2):

1. Describe the physical significance of wave function.
2. State HUP for position and momentum.
3. How does the size of a box affect the permitted energy levels of a particle?

Course Outcome 3 (CO 3):

1. Determine the number of quantum states in silicon between $(E_V - kT)$ and E_V at $T=300K$.
2. Describe the concept of Fermi level and its physical significance.
3. Calculate the probability that an energy state above E_F is occupied by an electron. Let $T= 300 K$. Determine the probability that an energy level $3kT$ above the Fermi energy is occupied by an electron.

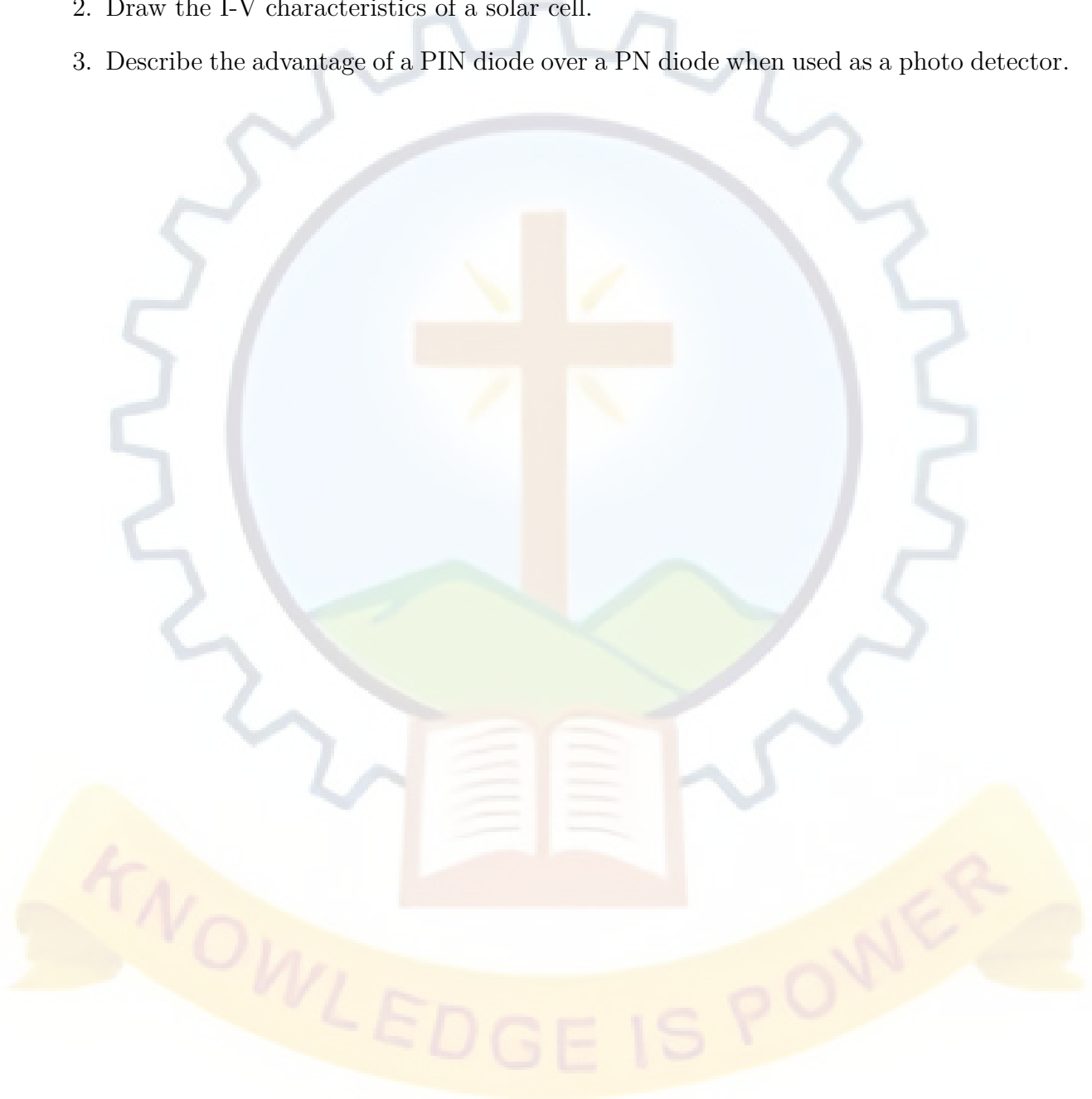
Course Outcome 4 (CO 4):

1. Describe the variation of Fermi level with temperature in an extrinsic semiconductor.
2. Determine the Fermi level and the max. doping concentration for which the Boltzmann approximation is still valid.

3. Sketch a graph of n_0 versus temperature for an n-type material.

Course Outcome 5 (CO 5):

1. Describe the formation of the depletion region.
2. Draw the I-V characteristics of a solar cell.
3. Describe the advantage of a PIN diode over a PN diode when used as a photo detector.



MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SECOND SEMESTER B.TECH DEGREE EXAMINATION, JUNE 2025

Course Code: B24PH1T01A

Course Name: ENGINEERING PHYSICS (A)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Explain the term population inversion.
2. Describe the principle of operation of optic fibers.
3. State Heisenberg's Uncertainty principle for conjugate variables.
4. Write the time independent Schrodinger equation.
5. What is meant by Fermi level?.
6. Write the Fermi-Dirac distribution function.
7. What are the factors affecting the Fermi level in an extrinsic semiconductor?
8. Plot the variation in E_F against donor concentration in an n-type semiconductor.
9. Sketch the energy bands in an unbiased, reverse-biased and forward-biased PN junction.
10. What is meant by a space charge region?

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Define numerical aperture of an optic fiber and derive an expression for the NA of a step index fiber with a neat diagram. 10
(b) Calculate the numerical aperture and acceptance angle of a fiber with a core refractive index of 1.54 and a cladding refractive index of 1.50 when the fiber is inside water of refractive index 1.33. 4

OR

12. (a) Outline the construction and working of Ruby laser. 10
(b) Calculate the N.A. of an optic fiber having core index of 1.54 and cladding index of 1.5. 4
13. (a) Derive time dependent Schrodinger equation. 10
(b) An electron is confined to a one dimensional potential box of length $2\text{\AA}$. Calculate the energies corresponding to the first and second quantum states in eV. 4

OR

14. (a) Derive the expression for the energy eigenvalues for a particle confined within a box of width L. 10
(b) Find the de-Broglie wavelength of an electron whose kinetic energy is 15eV. 4
15. (a) Derive the equations for the thermal equilibrium concentrations of electrons and holes in terms of the Fermi energy. 10
(b) Calculate the density of states per unit volume with energies between 0 eV and 1 eV. 4

OR

16. (a) Derive the equation for the intrinsic carrier concentration. 10
(b) Let $T=300\text{ K}$. Determine the probability that an energy level $3kT$ above the Fermi energy is occupied by an electron. 4
17. (a) Derive the fundamental relationship $n_0 p_0 = n_i^2$. 10
(b) Consider silicon at $T=300\text{ K}$ and assume that $N_c = 2.8 \times 10^{19} \text{ cm}^{-3}$ and $N_v = 1.04 \times 10^{19} \text{ cm}^{-3}$. Assuming that the Fermi energy is 0.25 eV below the conduction band and that the bandgap energy of silicon is 1.12 eV, determine the type of silicon under consideration. 4

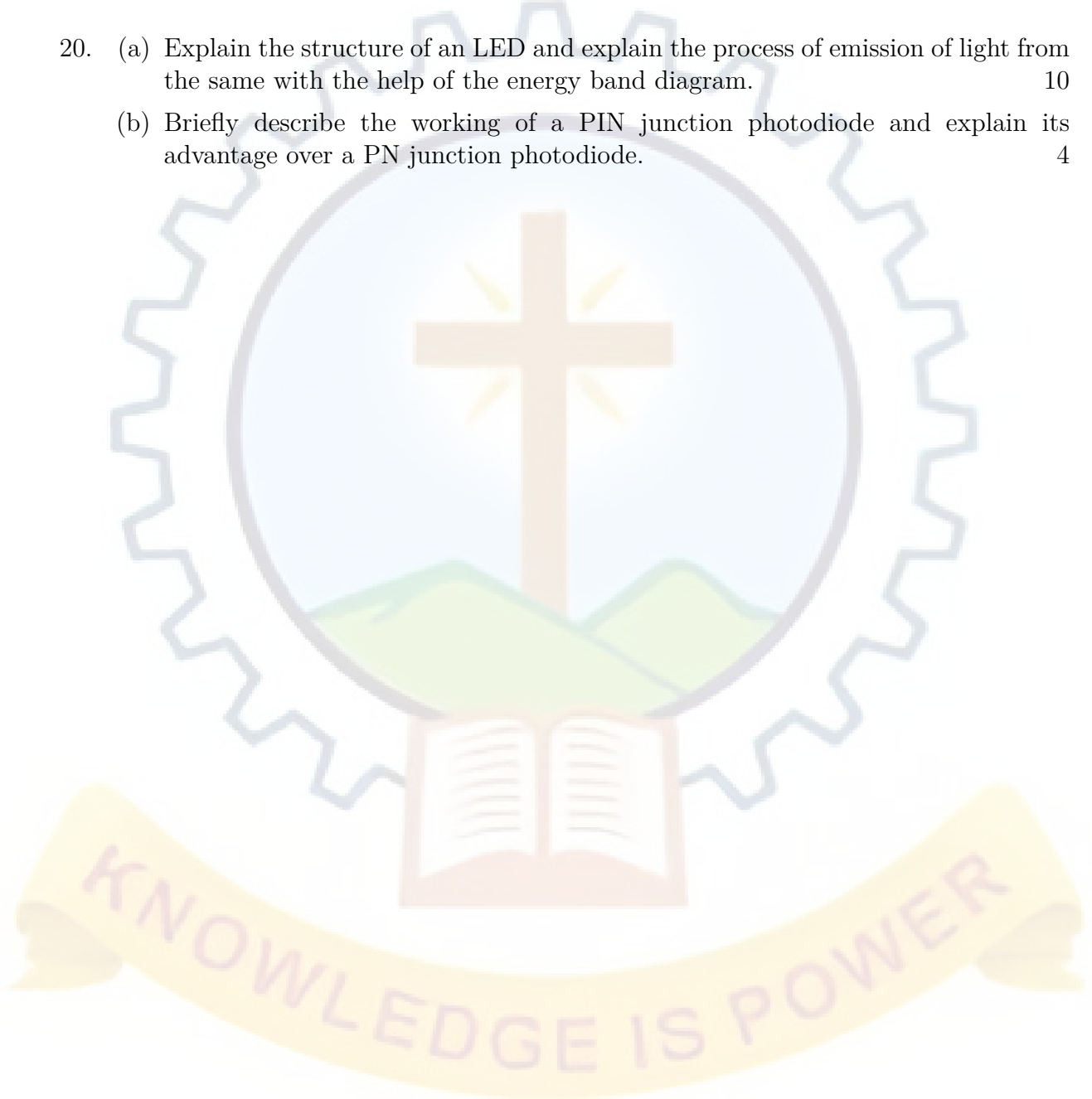
OR

18. (a) Derive the equations for n_0 and p_0 in terms of impurity doping concentrations. 10
(b) Silicon at $T=300\text{ K}$ contains an acceptor impurity concentration of $N_a = 10^{16} \text{ cm}^{-3}$. Determine the concentration of donor impurity atoms that must be added so that the silicon is n type and the Fermi level is 0.20 eV below the conduction-band edge. 4

19. (a) Describe the structure of energy bands in a PN junction under zero bias, forward bias and reverse bias and explain why conduction is possible only when it is forward-biased. 10
- (b) Write the ideal diode equation and draw the corresponding I-V characteristics. 4

OR

20. (a) Explain the structure of an LED and explain the process of emission of light from the same with the help of the energy band diagram. 10
- (b) Briefly describe the working of a PIN junction photodiode and explain its advantage over a PN junction photodiode. 4



B24CY1T01A	ENGINEERING CHEMISTRY (A)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4		2024

Preamble:

This basic science course will assist the students to acquire understanding in the concepts of chemistry for engineering applications and to familiarize the students with different application oriented topics like electrochemistry, nanomaterials, energy production, energy storage, OLED etc. Moreover, the students will be able to know analytical methods like various spectroscopic techniques, SEM etc. This will empower them to develop abilities and skills that are relevant to the study and practice of chemistry in their respective field of engineering.

Prerequisites: NIL

Course Outcomes:

After the completion of the course the student will be able to:

CO 1	Develop a comprehensive understanding of nanoscale materials, including their synthesis, fundamental properties and diverse applications(Cognitive Knowledge Level: Apply)
CO 2	Understand the principles and applications of various spectroscopic techniques and microscopic techniques such as SEM. (Cognitive Knowledge Level: Apply)
CO 3	Demonstrate an inclusive understanding of the principles of electrochemistry and corrosion. Also gain knowledge about various corrosion control methods. (Cognitive Knowledge Level: Apply)
CO 4	Learn about the basics of energy harvesting methods and its application. Apply the knowledge of battery, hydrogen generation and fuel cells in engineering. (Cognitive Knowledge Level: Apply)
CO 5	Apply the knowledge of conducting polymers and advanced materials in engineering. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO 1	2	1	1	1	1	1			1		1
CO 2	2	2		2	1				1		1
CO 3	2	1			1				1		1
CO 4	2	2	1		1				1		1
CO 5	2	1	1	1	1	1			1		1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (% Marks)	Test 2 (% Marks)	
Remember	30	30	30
Understand	50	50	50
Apply	20	20	20
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contains 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Fundamentals of Nanomaterials

Introduction - Classification - Based on dimension and structural composition - Nanoscale materials - Introduction - Properties and applications of Quantum dots, Graphene and Carbon nanotubes (CNT) - General Properties and applications of nanomaterials - Synthesis of nanomaterials - Top-Down and Bottom-Up approaches - Physical methods of synthesis - Mechanical milling, Laser ablation and Sputtering - Chemical methods of synthesis - Sol-Gel, co-precipitation and reduction.

MODULE 2 (8 hours)

Spectroscopic and Microscopic Techniques

Introduction - Types of spectrum - Electromagnetic spectrum - Molecular energy levels - Beer-Lambert's law - Numerical problems based on Beer-Lambert's law - Electronic spectroscopy (UV-vis) - Principle, instrumentation and applications - Types of electronic transitions - Vibrational spectroscopy (IR) - Principle and applications - Number of vibrational modes - Vibrational modes of CO_2 and H_2O - Force constant equation for diatomic molecules - Numerical problems based on force constant - Microscopic techniques - Scanning Electron Microscope (SEM) - Principle, instrumentation, working and applications.

MODULE 3 (7 hours)

Introduction to Electrochemistry and Corrosion Science

Introduction - Reference electrodes - Calomel electrode - Construction and working - Electrochemical series - Applications - Nernst equation for single electrode and cell (Derivation not required) - Applications - Effect of temperature on emf - Numerical problems based on Nernst equation - Corrosion - Introduction - Galvanic series - Types of corrosion - Galvanic and pitting corrosion - Corrosion control methods - Cathodic protection - Sacrificial anodic protection and impressed current cathodic protection - Electroplating of Copper - Electroless plating of Copper - Anodizing of Aluminium.

MODULE 4 (7 hours)

Energy Storage and Harvesting Technologies

Cells and batteries - Primary and secondary cells - Na-ion battery and Li-ion battery - Construction, working, advantages and applications - Hydrogen generation - Electrolysis of water - Fuel cells - Introduction - Construction and advantages of H_2 - O_2 fuel cell, Phosphoric acid fuel cell and Polymer Electrolyte Membrane Fuel Cell (PEMFC) - Supercapacitors - Classification - Construction and applications in hybrid vehicles.

MODULE 5 (7 hours)

Advanced Materials and Devices for Engineering Applications

Conducting polymers – Introduction - Classification - Intrinsically and extrinsically conducting polymers - Conduction mechanism – Band theory - Polyaniline and polypyrrole - Synthesis, properties and applications – Molecular devices based on conducting polymers – Diodes, Field Effect Transistor and Actuators - Introduction and applications - OLED – Construction, working and advantages - Smart materials - Thermo and light responsive materials - Introduction and examples - Sensors – Physical, chemical and biosensors – Introduction and applications.

Text Books

1. Jain and Jain, “Engineering Chemistry”, Dhanpat Rai Publishing Company, 17th edition 2015.
2. Shashi Chawla, “A Textbook of Engineering Chemistry”, Dhanpat Rai and Co. (P) Limited, 2017.
3. Muhammed Arif, Annette Fernandez, Kavitha P. Nair, “Engineering Chemistry”, Owl Books, 2019.
4. Ahad J., “Engineering Chemistry”, Jai Publication, 2019.
5. Roy K. Varghese, “Engineering Chemistry”, Crown Plus Publishers, 2019.
6. Soney C. George, Rino Laly Jose, “Text Book of Engineering Chemistry”, S. Chand and Company Pvt. Ltd., 2019.
7. B. L. Tembe, Kamaluddin, M. S. Krishnan, “Engineering Chemistry (NPTEL Web Book)”, 2018.

Reference Books

8. T. Pradeep, “NANO: The Essentials: Understanding Nanoscience and Nanotechnology”, McGraw-Hill, 2008.
9. B. Rogers, J. Adams, S. Pennathur, “Nanotechnology: Understanding Small Systems”, CRC Press, 2014.
10. Donald L. Pavia, “Introduction to Spectroscopy”, Cengage Learning India Pvt. Ltd., 2015.
11. J. Goldstein, “Scanning Electron Microscopy and Microanalysis”, Springer, 2012.
12. H. H. Willard, L. L. Merritt, “Instrumental Methods of Analysis”, CBS Publishers, 7th Edition, 2005.
13. Samuel Glasstone, “An Introduction to Electrochemistry”, East-West Press Pvt. Ltd., 2006.
14. Pietro Pedferri, “Corrosion Science and Engineering”, Springer Link, 2018.

15. B. Sunden, "Hydrogen, Batteries and Fuel Cells", Elsevier Inc., 2019.
16. B. Sorensen and G. Spazzafumo, "Hydrogen and Fuel Cells - Emerging Technologies and Applications", Elsevier Ltd., 2018.
17. Raymond B. Seymour, Charles E. Carraher, "Polymer Chemistry: An Introduction", Marcel Dekker Inc; 4th Revised Edition, 1996.
18. J. Janata, "Principles of Chemical Sensors" Springer, New York, NY, 2009.
19. F-G. Banica, "Chemical Sensors and Biosensors: Fundamentals and Applications", John Wiley and Sons, 2012.
20. M. Schwartz, "Smart Materials", CRC Press, 2008.
21. Y. Zhao, T. Ikeda, "Smart Light-Responsive Materials", Wiley, 2009.
22. V. Khutoryanskiy, T. Georgiou, "Temperature-Responsive Polymers: Chemistry, Properties and Applications", Wiley, 2018.
23. P. W. Atkins, "Physical Chemistry", Oxford University Press, 10th edn., 2014.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Fundamentals of Nanomaterials		7 hours
1.1	Introduction - Classification - Based on dimension and structural composition.	1
1.2	Nanoscale materials – Introduction - Properties and applications of Quantum dots, Graphene and Carbon nanotubes (CNT) – General properties and applications of nanomaterials.	3
1.3	Synthesis of nanomaterials - Top-Down and Bottom-Up approaches – Physical methods of synthesis - Mechanical milling, Laser ablation and Sputtering - Chemical methods of synthesis – Sol-Gel, co-precipitation and reduction.	3
Module 2: Spectroscopic and Microscopic Techniques		8 hours
2.1	Introduction - Types of spectrum - Electromagnetic spectrum - Molecular energy levels - Beer-Lambert's law – Numerical problems based on Beer-Lambert's law.	3

2.2	Electronic spectroscopy (UV-vis) – Principle, instrumentation and applications - Types of electronic transitions - Vibrational spectroscopy (IR) – Principle and applications - Number of vibrational modes - Vibrational modes of CO_2 and H_2O – Force constant equation for diatomic molecules - Numerical problems based on force constant.	4
2.3	Microscopic techniques - Scanning Electron Microscope (SEM) - Principle, instrumentation, working and applications.	1
Module 3: Introduction to Electrochemistry and Corrosion Science		7 hours
3.1	Introduction - Reference electrodes - Calomel electrode - Construction and working - Electrochemical series - Applications – Nernst equation for single electrode and cell (Derivation not required) – Applications – Effect of temperature on emf - Numerical problems based on Nernst equation.	3
3.2	Corrosion – Introduction - Galvanic series - Types of corrosion – Galvanic and pitting corrosion - Corrosion control methods - Cathodic protection - Sacrificial anodic protection and impressed current cathodic protection.	2
3.3	Electroplating of Copper - Electroless plating of Copper – Anodizing of Aluminium	2
Module 4: Energy Storage and Harvesting Technologies		7 hours
4.1	Cells and batteries – Primary and secondary cells – Na-ion battery and Li-ion battery - Construction, working, advantages and applications.	2
4.2	Hydrogen generation – Electrolysis of water - Fuel cells – Introduction - Construction and advantages of H_2 - O_2 fuel cell, Phosphoric acid fuel cell and Polymer Electrolyte Membrane Fuel Cell (PEMFC).	3
4.3	Supercapacitors - Classification - Construction and applications in hybrid vehicles.	2
Module 5: Advanced Materials and Devices for Engineering Applications		7 hours
5.1	Conducting polymers – Introduction - Classification - Intrinsically and extrinsically conducting polymers - Conduction mechanism – Band theory - Polyaniline and polypyrrole - Synthesis, properties and applications.	3

5.2	Molecular devices based on conducting polymers – Diodes, Field Effect Transistors, and Actuators - Introduction and applications - OLED – Construction, working and advantages.	2
5.3	Smart materials - Thermo and light responsive materials - Introduction and examples - Sensors – Physical, chemical and biosensors – Introduction and applications.	2

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. What are carbon nanotubes? Give two applications.
2. Comment on the structure of graphene.
3. How nanomaterials are classified based on structural composition?

Course Outcome 2 (CO 2):

1. State Beer-Lambert's law.
2. Calculate and sketch the vibrational modes of CO_2 .
3. What are the limitations of SEM?

Course Outcome 3 (CO 3):

1. How equilibrium constant is determined using electrochemical series?
2. Write the representation and reactions of calomel electrode.
3. Give any two differences between electrochemical series and galvanic series.

Course Outcome 4 (CO 4):

1. Compare Na-ion and Li-ion batteries.
2. List the applications of PEM fuel cell.
3. Discuss the classification of super capacitors.

Course Outcome 5 (CO 5):

1. Explain the preparation and properties of polypyrrole.
2. Discuss the working of OLED.
3. Compare physical and chemical sensors.



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIRST SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2024

Course Code: B24CY1T01A

Course Name: ENGINEERING CHEMISTRY (A)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Discuss the properties and applications of quantum dots.
2. How carbon nanotubes are classified based on structure?
3. State Beer-Lambert's law.
4. List the important applications of IR spectroscopy.
5. Explain how galvanic series can be used in corrosion control?
6. What is calomel electrode? Give the reduction reaction.
7. How does a PEM fuel cell differ from the other types of fuel cells?
8. Distinguish between primary and secondary cells with examples.
9. Give example and explain the importance of light responsive smart materials.
10. What are biosensors? Give their applications.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain two methods of chemical synthesis for nanomaterials. 8
(b) Discuss the classification of nanomaterials based on dimension. 6

OR

12. (a) What are nanoscale materials? Give the properties and applications of quantum dots and graphene. 9
(b) Explain the sputtering method for the synthesis of nanomaterials. 5
13. (a) Explain the principle, instrumentation and working of SEM. 8
(b) Calculate the force constant of HCl molecule, if it shows IR absorption at 2138cm^{-1} . Given that atomic masses of hydrogen and chlorine are 1 u and 35 u respectively. 6

OR

14. (a) Illustrate the vibrational modes of CO_2 and H_2O . Justify its IR activity. 9
(b) Explain the various energy levels associated with a molecule. 5
15. (a) How electroless plating of copper is carried out? Give the procedure and reactions. 8
(b) Write the cell reactions and calculate the emf of the cell $\text{Cu}/\text{Cu}^{2+} (1\text{M}) // \text{Ag}^+ (0.01\text{M}) // \text{Ag}$ at 30°C . Given $E^0 \text{Cu}^{2+}/\text{Cu} = 0.34\text{ V}$ and $E^0 \text{Ag}^+/\text{Ag} = 0.8\text{V}$. 6

OR

16. (a) What is cathodic protection? Explain two methods. 9
(b) Write the Nernst equation for Daniel cell and explain the effect of temperature on emf. 5
17. (a) Discuss the construction, working and advantages of Li-ion battery. 9
(b) What is electrolysis of water? 5

OR

18. (a) With a neat diagram explain the construction and working of Hydrogen-Oxygen fuel cell. 8
(b) Explain the classification of supercapacitors. 6
19. (a) Discuss the construction and working of OLED with a diagram. 9
(b) Explain the synthesis, properties and applications of polyaniline. 5

OR

20. (a) Elaborate the classification and applications of conducting polymers. 8
(b) What are smart materials? Give examples for heat responsive materials. 6

B24ES1T02	BASICS OF ELECTRICAL AND ELECTRONICS ENGINEERING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	2	0	5	4	2024

Preamble

This course aims to (1) fundamentals of circuit analysis, electrical components, machines, power systems, and safety practices (2) provide an overview of evolution of electronics, and introduce the working principle and examples of fundamental electronic devices and circuits (3) provide an introduction to digital electronics. Completing the course, students gain the necessary knowledge for more advanced courses and practical applications

Prerequisites

Physics and Mathematics (Pre-university level)

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Understand the essential circuit components and the fundamental circuit laws governing electrical circuits. (Cognitive Knowledge Level: Understand)
CO 2	Recall the basics of electromagnetism and the fundamentals of electrical machines and three-phase systems. (Cognitive Knowledge Level: Remember)
CO 3	Apply the basic knowledge of household wiring components and analyze electrical wiring layout for small residential buildings. (Cognitive Knowledge Level: Apply)
CO 4	Identify the active and passive electronic component and their specifications (Cognitive Knowledge Level: Understand)
CO 5	Design and analyze Rectifiers and Voltage amplifiers (Cognitive Knowledge Level: Apply)
CO 6	Explain the elements of digital system abstractions such as digital representations of information, digital logic and Boolean algebra (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	3	2	1		1	1	1	1	1	1
CO 2	3	3	2	1		1	1	1	1	1	1
CO 3	3	2	2	1		1	1	1	1	1	1
CO 4	2	1	1								1
CO 5	2	1	1								1
CO 6	2	1	1								1

Assessment Pattern

	BASIC ELECTRICAL			BASIC ELECTRONICS		
Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (% Marks)	Test 2 (% Marks)		Test 1 (% Marks)	Test 2 (% Marks)	
Remember	20	20	20	20	20	20
Understand	30	30	30	30	30	30
Apply	50	50	50	50	50	50
Analyse						
Evaluate						
Create						

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part I – Basic Electrical Engineering and Part II – Basic Electronics Engineering. Part I and PART II carries 50 marks each. For the end semester examination, part I contain 2 parts - Part A and Part B. Part A contain 5 questions carrying 4 marks each (not exceeding 2 questions from each module). Part B contains 2 questions from each module out of which one to be answered. Each question carries 10 mark and can have maximum 2 subdivisions. The pattern for end semester examination for part II is same as that of part I.

SYLLABUS

MODULE 1 (7 hours)

DC Electric Circuits

Passive components - R, L, and C, Sources - current and voltage sources, Resistances in series and parallel, current and voltage division rule, Ohm's Law, Kirchoff's Laws (Numerical problems).

Alternating Current Fundamentals

Generation of single-phase voltage - frequency, time period, average value, RMS value (sine wave concept only), Form and peak factors-Phasor representation of R,L,C, RL, RC, and RLC circuits - concept of impedance, power - active, reactive, and apparent, power factor (Numerical problems).

MODULE 2 (8 hours)

DC Machines and Transformers

Faraday's laws, Lenz's law, statically and dynamically induced EMF. DC Generator- construction and working principle, types, applications. DC motor - working principle, types of DC motors, applications. Transformer (single-phase only) - Construction, types-Working principle. Construction types

Three-Phase AC Systems

Generation of three-phase voltages - phase sequence, Y- Δ connection (balanced only),

relation between line and phase quantities, three-phase power, Single line diagram of a power system from generation to distribution.

MODULE 3 (8 hours)

Electrical wiring design

Electrical wiring system in domestic building - types of wiring, cables, Conduits, Switches and Outlets, switch boards, and distribution boards. Common power ratings of domestic gadgets, Codes and standards- Salient features of NEC, NBC and IE rule, NEC Symbols used in electrical wiring layout. Electrical lay out (single line diagram) for low- class domestic installation. Electrical load calculation- connected load method (Numerical problems).

Electrical Installation in Buildings

Protection devices - MCB, MCCB, ELCB/RCCB and RCBO- Principle of Operation-Rating and Specification, fuses-working and types. Electrical hazards and safety Precautions-Earthing & need of earthing, types.

MODULE 4 (8 hours)

Introduction to Electronics Components

Overview of the Evolution and Applications of Electronics. Familiarization of basic electronic components: Resistors, Capacitors, Inductors: Types, Specifications, Standard values, Color Coding.

Introduction to Semiconductor devices

Understanding PN Junction diode: Structure and Principle of Operation, V-I Characteristics, Diode Current equation (Simple problems), Special Diode: Zener Diode, Breakdown mechanisms, Bipolar junction Transistor: NPN and PNP Structure, Principle of operation of NPN Transistor

MODULE 5 (7 hours)

Introduction to Basic electronic circuits:

Rectifiers and power supplies: Block diagram description of a DC power supply, working of a full-wave bridge rectifier, capacitor filter (no analysis), working of simple Zener voltage regulator, Input and Output Characteristics of Common Emitter Configuration, Amplifier: RC-coupled amplifier using Voltage divider bias- Frequency Response-Bandwidth

MODULE 6 (7 hours)

Introduction to Digital Electronics

Number Systems: Decimal, Binary, Octal, and Hexadecimal number systems, Number Base Conversions, Binary Arithmetic: Addition, Subtraction, Multiplication, Logic gates, Universal Gates, Truth table, Realization of NOT gate using transistor

Text Books

1. D. P. Kothari and I. J. Nagrath, "Basic Electrical Engineering," 3rd Edition, Tata McGraw Hill."Electric Circuits & Networks", Pearson Education, 2009.
2. J. B. Gupta, "Theory and Performance of Electrical Machines" 15th Edition, S. K. Katarina & Sons.
3. M.K. Giridharan, Electrical System Design.
4. Chinmoy Saha, Arindham Halder and Debarati Ganguly, Basic Electronics - Principles and Applications, Cambridge University Press, 2018.
5. M.S.Sukhija and T.K.Nagsarkar, Basic Electrical and Electronics Engineering, Oxford University Press, 2012.

Reference Books

1. C. L. Wadhwa, "Basic Electrical Engineering," 4th Edition, New Age International Publisher
2. V. N. Mittle, "Basic Electrical Engineering," Tata McGraw Hill.
3. V. K. Mehta & Rohit Mehta, "Principles of Electrical Engineering," 6th Edition, S. Chand & Co. PVT. LTD.
4. S. K. Bhattacharya, "Basic Electrical and Electronics Engineering," 2nd Edition, Pearson Education.
5. D C Kulshreshtha, "Basic Electrical Engineering," 2nd Edition Tata McGraw Hill.
6. Del Toro V, "Electrical Engineering Fundamentals," 2nd Edition, New Delhi Prentice Hall of India.
7. Hughes, "Electrical and Electronic Technology", 10th Edition, Pearson Education.
8. R. K. Rajput, "Basic Electrical Engineering," 2nd Edition, Laxmi Publications PVT. LTD
9. Anant Agarwal, Jeffrey Lang, Foundations of Analog and Digital Electronic Circuits, Morgan Kaufmann Publishers, 2005.
10. Bernard Grob, Basic Electronics, McGraw Hil

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: DC Electric Circuits & Alternating Current Fundamentals		7 hours
1.1	DC Electric Circuits: Passive Components - R, L, and C, sources - current and voltage sources	1
1.2	Resistances in series and parallel, current and voltage division rule (Numerical problems).	1
1.3	Ohm's Law, Kirchoff's Laws (Numerical problems).	2
1.4	Alternating Current Fundamentals: Representation of sinusoidal waveforms - frequency, time period, average value, RMS value.	1
1.5	Phasor representation of R, RL, RC, RLC circuits - concept of impedance, power - active, reactive and apparent, power factor (Numerical problems).	2
Module 2: DC Machine and Three-Phase AC Systems		8 hours
2.1	Electromagnetic Induction: Faraday's laws, Lenz's law, statically and dynamically induced EMF	1
2.2	DC Machines: Construction and working principle - DC Generator – Types-applications.	2
2.3	DC motor - Construction and working principle-Types-applications	1
2.4	Transformers (single phase only): Working principle.	1
2.5	Three-Phase AC Systems: Generation of three-phase voltages - phase sequence.	1
2.6	Y- Δ connection (balanced only), relation between line and phase quantities, three-phase power.	2
Module 3: Electrical wiring design & Electrical Installation in Buildings		8 hours
3.1	Electrical wiring design: Electrical wiring system in domestic building - types of wiring, cables, Conduits, Switches and Outlets, switch boards, and distribution boards.	1

3.2	Common power ratings of domestic gadgets, Codes and standards- Salient features of NEC, NBC and IE rule, NEC Symbols used in electrical wiring layout.	1
3.3	Electrical lay out (single line diagram) for low- class domestic installation. Electrical load calculation-connected load method (Numerical problems).	2
3.4	Electrical Installation in Buildings: Protection devices - MCB, MCCB, ELCB/RCCB and RCBO- Principle of operation, fuses-working and types	2
3.5	Electrical hazards and safety precautions-Earthing & need of earthing, types, Electrical Safety & Precautions	2
3.6	Average value, rms value, form and peak factors of trapezoidal and sinusoidal waveforms - Numerical problems.	2
3.7	Phasor representation of sinusoidal quantities - phase difference, addition and subtraction of sinusoids.	1
3.8	Symbolic Representation: cartesian, polar and exponential forms.	1
Module 4: Introduction to Semiconductor devices		8 hours
4.1	Overview of the Evolution and Applications of Electronics.	1
4.2	Familiarization of basic electronic components: Resistors, Capacitors, Inductors: Types, Specifications, Standard values, Color Coding.	3
4.3	Understanding PN Junction diode: Structure and Principle of Operation	1
4.4	V-I Characteristics, Diode Current equation (Simple problems)	1
4.5	Special Diode: Zener Diode, Breakdown mechanisms	1
4.6	Bipolar junction Transistor: NPN and PNP Structure, Principle of operation of NPN Transistor	1
Module 5: Introduction to Basic electronic circuits		7 hours
5.1	Rectifiers and power supplies: Block diagram description of a dc power supply, Working of a full wave bridge rectifier, capacitor filter (no analysis), working of simple Zener voltage regulator.	4
5.2	Input and Output Characteristics of Common Emitter Configuration	1

5.3	Amplifier: RC Coupled Amplifier using Voltage divider bias- Frequency response-Bandwidth .	2
Module 6: Introduction to Digital Electronic		7 hours
6.1	Number Systems: Decimal, Binary, Octal, and Hexadecimal number systems, Number Base Conversions	2
6.2	Binary Arithmetic: Addition, Subtraction, Multiplication	2
6.3	Logic gates, Universal Gates, Truth table .	2
6.4	Realization of NOT gate using transistor .	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. Solve problems based on series and parallel circuits.
2. Solve problems based on current and voltage division rules.
3. Solve problems using Kirchoff's laws.
4. Phasor representation of R, RL, RC and RLC circuits.
5. Problems on rms and average values of periodic waveforms.
6. Problems related to power and power factor.

Course Outcome 2 (CO 2):

1. Construction and working of DC generator and DC motor.
2. Different types and applications of DC generator and DC motor.
3. Working principle of single-phase transformer.
4. Problems on three-phase line & phase quantities for a balanced load.

Course Outcome 3 (CO 3):

1. Electrical wiring system in domestic building.
2. Codes and standards
3. Electrical lay out (single line diagram).

4. Electrical load calculation- connected load method (Numerical problems).
5. Protection devices and its principle of operation.
6. Electrical hazards and safety Precautions-Earthing & need of earthing, types, Electrical Safety & Precautions.

Course Outcome 4 (CO 4):

1. Explain the significance of color coding in identifying the values of resistors.
2. Describe the structure of a PN junction diode and its principle of operation.
3. Describe the structure of a PN junction diode and its principle of operation.

Course Outcome 5 (CO 5):

1. What is the need of voltage divider biasing in an RC coupled amplifier?
2. Analyze the importance of selection of operating point in the context of a BJT amplifier.
3. Why is it required to have a voltage amplifier in a public address system?

Course Outcome 6 (CO 6):

1. Convert 203:5210 to binary and hexadecimal.
2. Implement an AND gate using NOR gate.
3. Define logic gates and explain their fundamental role in digital electronics.

MODEL QUESTION PAPER

QP CODE:

Pages: 4

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIRST SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2024

Course Code: : B24ES1T02

***Course Name: BASICS OF ELECTRICAL AND ELECTRONICS
ENGINEERING***

Max. Marks: 100

Duration: 3 hours

PART 1: BASICS OF ELECTRICAL ENGINEERING

PART A

Answer all questions. Each question carries 4 marks.

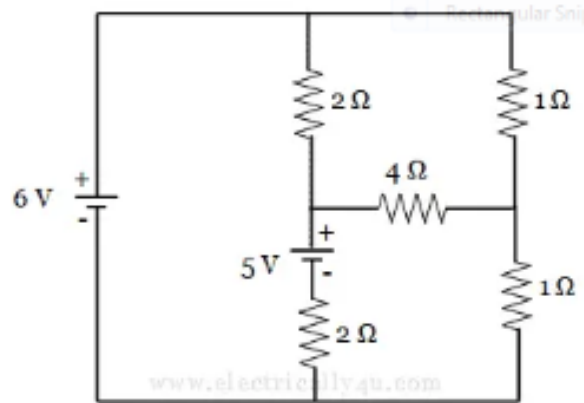
1. State and explain Kirchhoff's laws with examples.
2. Differentiate between statically and dynamically induced emf.
3. Derive the relation between line and phase current in a 3-phase delta-connected system.
4. Distinguish between MCB and MCCB.
5. What is the need for earthing? Describe the different types of earthing.

PART B

Answer any one full question . Each question carries 10 marks.

6. For the circuit shown below, determine the current flows through all the resistors using Kirchhoff's law. 10

OR



7. A resistance of 10Ω and inductance of 0.3H and a capacitance of $100\mu\text{F}$ are connected in series across 230V , 50Hz single-phase supply. Calculate the 10
- (a) Impedance of the circuits
 - (b) Current through the circuits
 - (c) Voltage across R, L, and C
 - (d) Power consumed by the circuit.
8. A 3-phase, 400V , 4 wire system has a balanced star connected load with impedance $Z = 15 + j10\Omega$ each. Find the line currents and the total power consumed by the load. 10

OR

9. (a) State Faraday's laws of electromagnetic induction. 4
(b) Explain the construction and working principle of DC motor. 6
10. What is the role of NEC and NBC in building design? 10
- OR**
11. (a) Explain the different types of wiring. 5
(b) What are the different NEC symbols used in electrical wiring layout? 5

PART 2: BASICS OF ELECTRONICS ENGINEERING

PART A

Answer all questions. Each question carries 4 marks.

1. (a) Identify the colour code for the given resistor values.
 - i. $1 + 5\%$
 - ii. $3.3k + 1\%$(b) Identify the capacitor value with unit.
2. Explain the break down mechanisms of Zener diode.
3. Briefly Discuss the block diagram of a DC power supply.
4. For a NPN transistor $\alpha = 0.98$ and $I_B = 100 \mu A$. Find I_E and I_C .
5. Which gates are called universal gates and why?

PART B

Answer any one full question . Each question carries 10 marks.

6. (a) Explain with necessary diagrams the principle of operation of NPN transistor. 5
(b) Write the diode current equation, If the reverse saturation current of Germanium diode at room temperature is 0.4 micro ampere. Determine the current flowing through the diode when 0.2V is applied at room temperature. 5

OR

7. (a) Plot and explain the V-I characteristics of a PN junction Diode. 5
(b) Describe the color coding of a resistor with suitable example. 5
8. With necessary diagrams explain the working of a full wave bridge rectifier. 10

OR

9. Describe the input and output characteristics of Common emitter configuration. 10
10. Convert the following numbers to binary. 10
 - (a) EE_{16}
 - (b) $FD654_{16}$
 - (c) 33_{10}
 - (d) 17_{10}
 - (e) 1142_8

OR

11. Draw the symbol and truth table of AND, OR, NAND, NOR and XOR.

10

B24ES1L02	BASIC ELECTRICAL AND ELECTRONICS WORKSHOP	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	2	1		2024

Preamble

The course aims to 1) impart fundamentals of electrical wiring, safety measures, and troubleshooting to students. 2) expose students to the concepts of various wiring methods and distribution systems. 3) gives the basic introduction of electronic hardware systems and provide hands-on training with familiarization, identification, testing, assembling, dismantling, fabrication and repairing such systems by making use of the various tools and instruments available in the Electronics Workshop.

Prerequisite

Nil

Course Outcomes

After the completion of the course the student will be able to

CO 1	Identify electrical symbols, measuring instruments, accessories, and tools used for electrical wiring. (Cognitive Knowledge Level: Apply)
CO 2	Understand the substation, distribution system, and safety measures against electrical shocks and select the fuse unit for a given electrical circuit. (Cognitive Knowledge Level – Understand)
CO 3	Estimate and develop the electric circuits for wiring domestic and industrial buildings. (Cognitive Knowledge Level – Apply)
CO 4	Demonstrate proficiency in identifying various electronic components, including active, passive, electrical, electronic, and electromechanical components (Cognitive Knowledge Level-Understand)
CO 5	Develop and illustrate electronic circuit diagrams using recognized standards such as BIS/IEEE symbols and utilize Electronic Design Automation (EDA) tools for schematic capture and simulation. (Cognitive Knowledge Level-Apply)
CO 6	Design and fabricate electronic circuits on boards, troubleshooting of minor problems in electronic equipment and handling of test and measuring equipment (Cognitive Knowledge Level-Apply)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	2	3	1		3	1	3	3	2	3
CO 2	3	2	3	1		3	1	3	3	2	3
CO 3	3	3	3	3		3	1	3	3	3	3
CO 4	3	1	1	1				1	1		2
CO 5	3	2	2	1	2			2	1		2
CO 6	3	2	2	1	2			2	1		1

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	70	30	1 hour

Continuous Internal Evaluation Pattern

Attendance	20 marks
Class Work/ Assessment Viva Voce	50 marks
Viva Voce / test	30 marks

End Semester Examination Pattern

The college will internally conduct end-semester examination. Separate ESE 's will be held for Electrical workshop and Electronics workshops, each in the form of a one-hour written/objective exam. The total marks for this course is equally divided between the Electrical and Electronics workshop.

SYLLABUS

LIST OF EXPERIMENTS PART I

ELECTRICAL

1	1. Familiarization with electrical symbols, measuring instruments, lighting and wiring accessories, tools, and various wiring systems. 2. Familiarization with earthing in electrical installations, precautions against the electric shock phenomenon and safety procedures.
2	Realization of domestic wiring 1. Wiring of one lamp controlled by one switch and a 3-pin plug socket controlled independently. 2. Wiring of one lamp controlled by two switches (Staircase wiring).
3	1. Realization of Industrial wiring - Wiring of three lamps controlled by three switches (Godown wiring). 2. Study of fuse, MCB, ELCB, RCCB and selection of fuse rating for circuits with medium and high power.
4	Wiring of the distribution board, including the power plug, an isolator, MCB, and ELCB for 1000 W power.
5	Measurement of low-medium-high resistance using the megger and voltmeter-ammeter method.
6	Visit the on-campus substation and familiarize with the supply system, transformer, HT Panel, and distribution system.

Reference Books

1. H Cotton, Advanced Electrical Technology, Reem Publications, 2011.
2. Suresh Kumar K.S, Electrical Circuit and Networks, Pearson Education, New Delhi, 2009.
3. EW. Golding, Electrical Measurements and Measuring Instruments, 5th ed. Reem Publications, 2011.
4. A course in electrical installation estimating and costing, J Bh Gupta, 9th edition, 2012

PART II ELECTRONICS

1	Familiarization/Identification of electronic components with specification (Functionality, type, size, colour coding, package, symbol, cost, etc. [Active, Passive, Electrical, Electronic, Electro-mechanical, Wires, Cables, Connectors, Fuses, Relays, Crystals, Displays, Heat sink, etc.) .
2	Drawing of electronic circuit diagrams using BIS/IEEE symbols and introduction to EDA tools (such as Dia, Xcircuit, LT SPICE).
3	Familiarization/Application of testing instruments and commonly used tools. [Multimeter, Function generator, Power supply, DSO etc.] [Soldering iron, De-soldering pump, Pliers, Cutters, Wire strippers, Screwdrivers, Tweezers etc.]
4	Testing of electronic components [Resistor, Capacitor, Diode, Transistor and JFET using multimeter.
5	Interconnection methods using breadboard and soldering practice. [Soldering - types - selection of materials and safety precautions, soldering practice in general purpose PCB/lug strip] .
6	Design and fabrication of a single sided PCB for a Fixed voltage power supply with transformer, rectifier, capacitor filter and Zener regulator. Assemble and test the circuit.

Reference Books

1. "Electronic Devices and Circuit Theory" by Robert L. Boylestad and Louis Nashelsky.
2. "Fundamentals of Electronic Circuits" by Charles K. Alexander and Matthew N.O. Sadiku.
3. "The Soldering Handbook" by M.W. Schwartz.
4. "Electronic Devices", by Thomas L.Floyd Ninth Edition.
5. "Basic electronics and linear circuits", by N.N.Bharagava, D.C.Kulshreshtha, S.C.Gupta, TataMcGraw-Hill,1994.

B24ES1L01A	PROGRAMMING LAB (A)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	3	2		2024

Preamble

The course aims to provide students with exposure to problem solving through C Programming. The students will have hands on experience in C programming, array handling, string manipulations, function creation, structure and pointer operations and file processing. After the lab sessions the student will be able to analyze complex problems and find solutions for real word problems.

Prerequisite

Nil

Course Outcomes

After the completion of the course the student will be able to

CO1	C programs with branching and looping statements for processing arrays and matrices. (Cognitive Knowledge Level: Apply)
CO2	Divide a given computational problem into a number of modules and develop functions to find the solutions to the computational problem and also create programs for string processing (Cognitive Knowledge Level: Apply)
CO3	Construct C programs for searching and sorting (Cognitive Knowledge Level: Apply)
CO4	Develop C programs which use structures and pointers for data processing and parameter passing(Cognitive Knowledge Level: Apply)
CO5	Develop C programs for file processing(Cognitive Knowledge Level:Apply)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	1	2	2				1		
CO 2	2	3	3	2	3	1			2	1	3
CO 3	2	3	2	2	3				1		
CO 4	2	3	3	2	3	1			2		1
CO 5	2	3	2	2	3				3		1

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Evaluation	20 marks
- Viva (Self Study)	10 marks
- Output	10 marks
Lab Record	5 marks
Viva-voce/Test	15 marks

SYLLABUS

LIST OF EXPERIMENTS

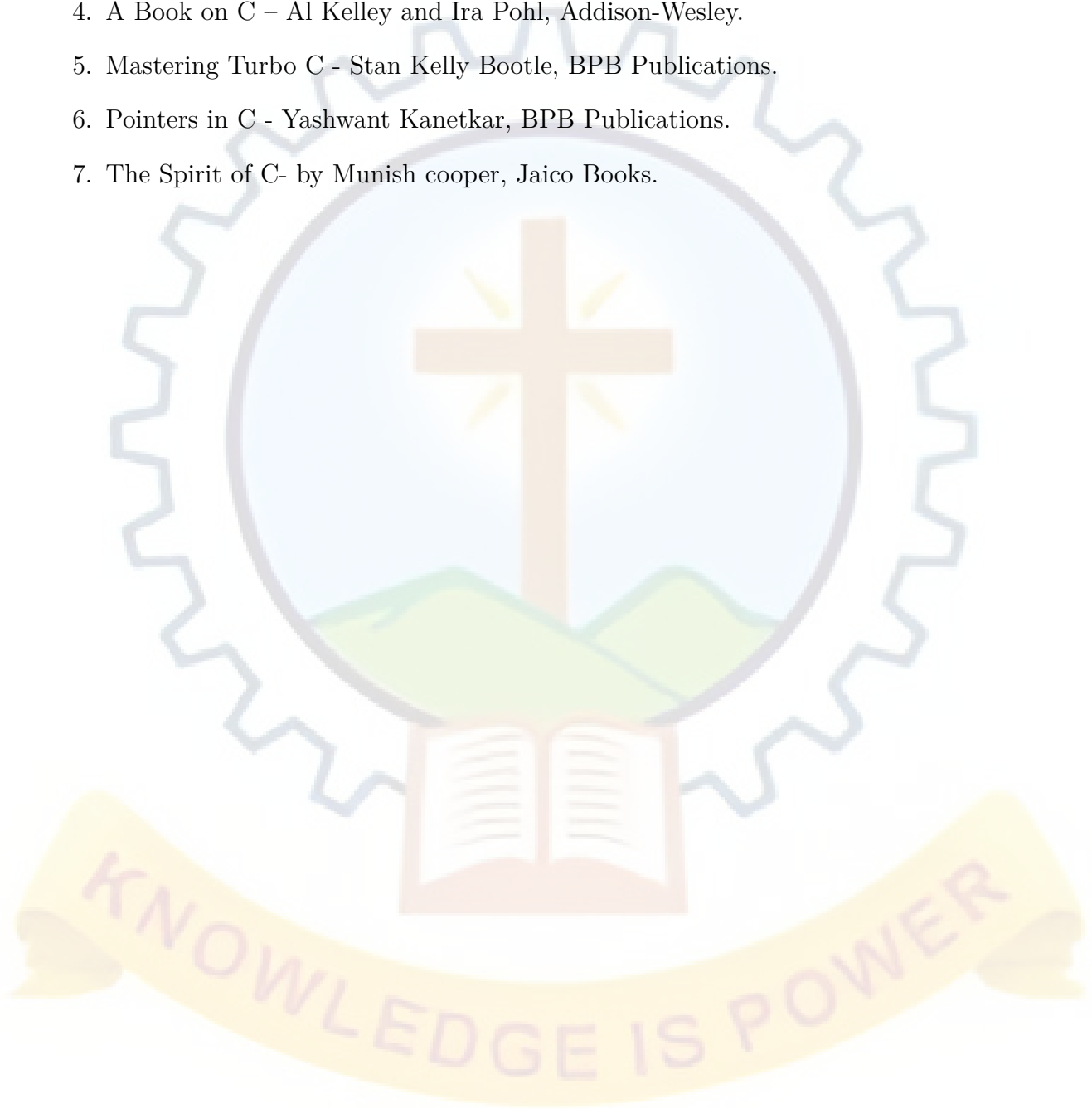
1	Familiarization of Linux Commands.
2	Familiarization of IO console. a) Write a C program to display the Personal details. b) Write a C program to Add two numbers c) Write a C program to evaluate the arithmetic expression using command line arguments
3	Familiarization of Operators. a) Write a C program to evaluate bitwise operations on given numbers. b) Write a C program to swap two numbers using XOR operation. c) Write a C program to find the largest of three numbers using conditional operator.

4	Write a C program for the salary increment of an employee (eg: if the given salary is > 50000 , 20 % increment)
5	Write a menu driven program to perform the Calculator operations, namely addition, subtraction, multiplication, division and square of a number.
6	Write a C program to check the given number is Armstrong or not and find the reverse of the number
7	Write a C program to find the sum of first N natural numbers using array.
8	Write a C program a) To read an array of size n and display in reverse order. b) Display the sum and average of the array elements.
9	Write a C program to read an array of size n and display the prime numbers in the array.
10	Write a C program to read n integers, store them in an array and search for an element in the array using Linear Search.
11	Write a C program to read n integers, store them in an array and sort the elements using Bubble Sort.
12	Write a C program to read a string (word), store it in an array and check whether it is a palindrome word or not.
13	Write a C program to read two strings (each one ending with a \$ symbol), store them in arrays and concatenate them without using library functions.
14	Write a C program to read a string (ending with a \$ symbol), store it in an array and count the number of vowels, consonants and spaces in it.
15	Write a C program to read a string (word), store it in an array and obtain its reverse by using a user-defined function.

16	Write a menu-driven program for performing matrix addition, multiplication and finding the transpose. Use functions to a) Read a matrix. b) Find the sum of two matrices. c) Find the product of two matrices d) Find the transpose of a matrix and e) Display a matrix.
17	Find the factorial of a given Natural Number n using recursive and non-recursive functions.
18	Write a C Program to find the largest of three numbers using Macros.
19	Using structure, read and print data of n employees (Name, Employee Id and Salary).
20	Write a C program to declare a union containing 5 string variables (Name, House Name, City Name, State and Pin code) each with a length of C_SIZE (user defined constant).Then, read and display the address of a person using a variable of the union.
21	Write a C program to read the student details using structure and display the information through the user defined function.
22	Do the following using pointers a) add two numbers. b) swap two numbers using a user defined function.
23	Read and display the elements of an array using pointers, Compute the sum of the elements stored in the array using pointers and user defined function.
24	Create a file and perform the following a) Write data to the file. b) Read the data in a given file & display the file content on console. c) Append new data and display on console
25	Open a text file and count the number of characters, words and lines in it; and store the result in another file.
26	Find the substring from the given text file and replace it with another string.

Reference Books

1. Programming in C - Stephen C. Kochan, CBS publishers.
2. Programming in C – E. Balaguruswamy , Mc Graw Hill.
3. Let us C – Yashwant Kanetkar, BPB Publications.
4. A Book on C – Al Kelley and Ira Pohl, Addison-Wesley.
5. Mastering Turbo C - Stan Kelly Bootle, BPB Publications.
6. Pointers in C - Yashwant Kanetkar, BPB Publications.
7. The Spirit of C- by Munish cooper, Jaico Books.



B24PC1L01A	PHYSICS(A) & CHEMISTRY (A) LAB	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	2	2	1	2024

Preamble

This integrated Physics-Chemistry laboratory course is designed to provide engineering students with essential hands-on experience and foundational experimental skills. Through a series of structured experiments, students will develop proficiency in precise measurement and accurate data recording. The course bridges the gap between theoretical concepts and practical applications by enabling learners to operate scientific instruments, perform fundamental chemical techniques and critically compare experimental results with theoretical predictions. This experiential learning course equips students to critically analyse and interpret data effectively and apply the physical and chemical principles to address complex engineering challenges.

Prerequisite

Nil

Course Outcomes

After the completion of the course the student will be able to

CO1	Develop analytical/experimental skills and impart prerequisite hands-on experience for engineering laboratories. (Cognitive Knowledge Level: Apply)
CO2	Understand the need for precise measurement practices for data recording. (Cognitive Knowledge Level: Apply)
CO3	Understand the principle, concept, working and applications of relevant technologies and compare results with theoretical calculations. (Cognitive Knowledge Level: Apply)
CO4	Understand and practice fundamental techniques in chemistry to generate experimental skills. (Cognitive Knowledge Level: Apply)
CO5	Learn to design and carry out scientific experiments as well as accurately record and analyze the results of such experiments. (Cognitive Knowledge Level: Apply)
CO6	Acquire the ability to understand different methods of chemical synthesis and instrumental techniques to solve various engineering problems. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2			1	1						
CO 2	1			1	1						
CO 3	2	1									
CO 4	1	1		1	1						
CO 5	1	1		3	1						
CO 6	2	1		1							

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	70	30	1 Hour

Continuous Internal Evaluation Pattern

Attendance	20 marks
Class Work/ Assessment Viva-Voce	50 marks
End semester examination (Internally by the college)	30 marks

End Semester Examination Pattern

The college will internally conduct the end semester examination. Separate ESE's will be held for the Physics Lab and Chemistry Lab, each in the form of a 30 minutes written objective examination. The total marks for this course will be equally divided between the Engineering Physics Lab and Engineering Chemistry Lab.

**SYLLABUS
LIST OF EXPERIMENTS**

**PART I
PHYSICS (A) LAB**

1	WDSO-Measurement of frequency and amplitude of waveforms.
2	Optic Fiber -Measurement of Splice Loss.
3	Junction Diode - Measurement of E_R .
4	Photoelectric cell - Calculation of Planck's constant.
5	Optic Fiber - Measurement of Numerical Aperture.
6	I-V characteristics of solar cells.
7	Optic Fiber - Measurement of Bending Loss.
8	LED Characteristics.

**PART II
CHEMISTRY (A) LAB**

(MINIMUM FOUR EXPERIMENTS ARE MANDATORY)

1	Determination of molar absorptivity of a compound.
2	Potentiometric redox titration.
3	Verification of Nernst equation using Daniel cell.
4	Determination of wavelength of absorption maximum and colorimetric estimation of Fe^{3+} ions in the solution.
5	Electroplating with copper.
6	Synthesis of iron oxide nanoparticles.
7	Estimation of sodium ions by flame photometry.
8	Synthesis of conducting polyaniline from aniline.

Reference Books

1. S.L. Gupta and Dr. V. Kumar, "Practical Physics with viva voice", Pragati Prakashan Publishers, Revised Edition, 2009.
2. M.N. Avadhanulu, A.A. Dani and Pokely P.M, "Experiments in Engineering Physics", S.Chand & Co, 2008.
3. S. K. Gupta, "Engineering physics practicals", Krishna Prakashan Pvt. Ltd., 2014 .
4. P. R. Sasikumar, "Practical Physics", PHI Ltd., 2011.
5. D.R Mehta, "Laboratory Manual Physics", D.K Publishing House.
6. G. Svehla, B. Sivasankar, "Vogel's Qualitative Inorganic Analysis", Pearson, 2012.
7. R. K. Mohapatra, "Engineering Chemistry with Laboratory Experiments", PHI Learning, 2017.
8. Muhammed Arif, "Engineering Chemistry Lab Manual", Owl publishers, 2019.
9. Roy K. Varghese, "Engineering Chemistry Laboratory Manual", Crown plus Publishers, 2019.
10. Soney C. George, Rino Laly Jose, "Lab Manual of Engineering Chemistry", S. Chand & Company Pvt Ltd., New Delhi, 2019.
11. S. M. Ashraf, "A Laboratory Manual of Polymers" I. K. International Publishing House Pvt. Ltd., 2008
12. Ulrich Schubert, Nicola Hüsing, "Material Synthesis: A Practical Guide", Springer Vienna, 2008.
13. Anu Tresa Sunny, Prajitha Velayudhan, Sabu Thomas, "Colloidal metal Oxide Nanoparticles: Synthesis, Characterization and Applications", Elsevier Science, 2019.

B24MC1T01	LIFE SKILLS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		1	0	1	2	P/F	2024

Preamble

This Course is aimed at equipping individuals with the essential competencies to navigate life's challenges with resilience and positivity. This course, embarks on a profound exploration of personal development, fostering self-awareness, meaningful connections, and the ability to navigate the complexities of both the abstract and the concrete aspects of life. It aims to enhance employability by providing practical insights and hands-on experiences that will empower one to apply these principles effectively in one's personal and professional endeavors.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Empower individuals with the knowledge and practical skills needed to navigate life challenges and to cope with emotions and stress. (Cognitive Knowledge Level: Apply)
CO 2	Develop a profound understanding of themselves and others, leading a fulfilling professional life by embracing a holistic approach to well being. (Cognitive Knowledge Level: Analyze)
CO 3	Provide a solid foundation in leadership principles and team dynamics. (Cognitive Knowledge Level: Apply)
CO 4	Basic understanding of financial concepts for financial well being. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	1	1				2	2	3	2	2	2
CO 2	1	1				2	3	3	2	2	2
CO 3	1	1				2	2	3		3	2
CO 4	2	1	2			2				3	

Assessment Pattern

Bloom's Category	Continuous Assessment	End Semester Examination (% Marks)
	Test (%Marks)	
Remember	20	20
Understand	20	20
Apply	30	30
Analyse	30	30
Evaluate		
Create		

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	50	50	2 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (1 number)	25 marks
Self Study (Regular Assessment)	15 marks

Regular assessment

Group Discussion (Marks: 9)

Create groups of about 6 students each and engage them on a GD on a suitable topic for about 20 minutes. Parameters to be used for evaluation are as follows:

1. Communication Skills: 3 marks
2. Subject Clarity: 2 marks

3. Group Dynamics: 2 marks
4. Behaviors & Mannerisms: 2 marks

Presentation Skills (Marks: 6)

Identify a suitable topic and ask the students to prepare a presentation (preferably a PowerPoint presentation) for about 10 minutes. Parameters to be used for evaluation are as follows

1. Communication Skills: 2 marks
2. Platform Skills: 2 marks
3. Subject Clarity/Knowledge: 2 marks

End Semester Examination Pattern

Part A: Short answer question (20 marks)

There will be one question from each MODULE (four questions in total, five marks each). Each question should be written in about maximum of 400 words. Parameters to be used for evaluation are as follows:

1. Content Clarity/Subject Knowledge
2. Presentation style
3. Organization of content

Part B: Case Study (30 marks)

The students will be given a case study with questions at the end. The students have to analyze the case and answer the question at the end. Parameters to be used for evaluation are as follows:

1. Analyze the case situation
2. Key players/characters of the case
3. Identification of the problem (both major & minor if exists)
4. Bring out alternatives
5. Analyze each alternative against the problem
6. Choose the best alternative
7. Implement as solution
8. Conclusion
9. Answer the question at the end of the case

SYLLABUS

MODULE 1 (6 hours)

Overview of Life Skills:

Meaning and significance of life skills, Life skills identified by WHO: Self-awareness, Empathy, Critical thinking, Creative thinking, Decision making, problem solving, Effective communication, interpersonal relationship, coping with stress- Four A's of stress management, Gratitude Training, Coping with emotion- PATH method and relaxation techniques.

MODULE 2 (6 hours)

Life Skills for Professionals:

positive thinking, right attitude, Experience, attention to detail, having the big picture, learning skills, research skills, setting goals and achieving them, perseverance, motivation, self-motivation, and motivating others, IQ, EQ, and SQ, Collaboration, continuous learning, unlearning and relearning, cross cultural communication, social media etiquettes, Financial Literacy.

Time Management: Prioritizing tasks, setting realistic goals and managing time effectively, work life balance.

Holistic Thinking: imagination, intuition, lateral thinking, Multiple intelligence, spirituality, family bonding, living peacefully.

MODULE 3 (6 hours)

Leadership:

Leadership traits, Styles of Leadership, VUCA Leadership, Transactional vs Transformational Leaders, managing diverse stakeholders, crisis management, Effective Leaders.

Group and Team Dynamics: Group vs Team, Team Dynamics, Virtual teams, managing team performance and managing conflicts, Intrapreneurship

MODULE 4 (6 hours)

Financial Literacy:

Time value of money, power of compounding, Future value of a single cash flow, effective versus nominal rate, Future value of an annuity, present value of a single cash flow, Present value of an annuity.

Reference Books

1. Shiv Khera, *You Can Win*, Macmillan Books, New York, 2003.
2. Barun K. Mitra, "Personality Development & Soft Skills", Oxford Publishers, Third impression, 2017.
3. ICT Academy of Kerala, "Life Skills for Engineers", McGraw-Hill Education (India) Private Ltd., 2016.
4. Caruso, D. R. and Salovey P, "The Emotionally Intelligent Manager: How to Develop and Use the Four Key Emotional Skills of Leadership", John Wiley & Sons, 2004.
5. Kalyana, "Soft Skill for Managers"; First Edition; Wiley Publishing Ltd, 2015.
6. Larry James, "The First Book of Life Skills"; First Edition, Embassy Books, 2016.
7. Shalini Verma, "Development of Life Skills and Professional Practice"; First Edition; Sultan Chand (G/L) & Company, 2014.
8. Daniel Goleman, "Emotional Intelligence"; Bantam, 2006.
9. Remesh S., Vishnu R.G., "Life Skills for Engineers", Ridhima Publications, First Edition, 2016.
10. Butterfield Jeff, "Soft Skills for Everyone", Cengage Learning India Pvt Ltd; 1 edition, 2011.
11. Training in Interpersonal Skills: Tips for Managing People at Work, Pearson Education, India; 6 editions, 2015.
12. The Ace of Soft Skills: Attitude, Communication and Etiquette for Success, Pearson Education; 1 edition, 2013
13. Prasanna Chandra, "Fundamentals of Financial Management", McGraw Hill Education (India) Private Ltd, 2020
14. Edward de Bono, "Lateral Thinking"
15. Howard Gardner, "Multiple Intelligences"

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		24 Hours
Module 1: Overview of Life Skills		6 hours
1.1	Overview of Life Skills: Meaning and significance of life skills, Life skills identified by WHO: Self-awareness, Empathy, Critical thinking, Creative thinking, Decision making	1
1.2	Problem solving, Effective communication, interpersonal relationship, coping with stress- Four A's of stress management.	1
1.3	Gratitude Training, Coping with emotion- PATH method and relaxation techniques	1
1.4	Activity- Presentation, Group discussion	3
Module 2: Life Skills for Professionals		6 hours
2.1	Life skills for professionals: positive thinking, right attitude, Experience, attention to detail, having the big picture, learning skills, research skills, setting goals and achieving them, perseverance, motivation, self-motivation, and motivating others,	1
2.2	IQ, EQ, and SQ, Collaboration, continuous learning, unlearning and relearning, cross cultural communication, social media etiquettes, Financial Literacy.	1
2.3	Time management: Prioritizing tasks, setting realistic goals and managing time effectively, work life balance.	1
2.4	Holistic Thinking: imagination, intuition, lateral thinking, Multiple intelligence, spirituality, family bonding, living peacefully.	1
2.5	Activity- Presentation, Group discussion.	2
Module 3: Leadership		6 hours
3.1	Leadership: Leadership traits, Styles of Leadership, VUCA Leadership, Transactional vs Transformational Leaders, managing diverse stakeholders, crisis management, Effective Leaders.	1

3.2	Group and Team Dynamics: Group vs Team, Team Dynamics, Virtual teams, managing team performance and managing conflicts, Intrapreneurship	1
3.3	Activity- Presentation, Group discussion	4
Module 4: Financial Literacy		6 hours
4.1	Financial Literacy: Time value of money, power of compounding, Future value of a single cash flow.	1
4.2	Effective versus nominal rate, Future value of an annuity.	1
4.3	Present value of a single cash flow, Present value of an annuity.	1
4.4	Activity- Presentation, Group discussion	3

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. What are the life skills identified by WHO?
2. List the 4 A's of stress management.
3. Differentiate between Critical thinking and Creative thinking.

Course Outcome 2 (CO 2):

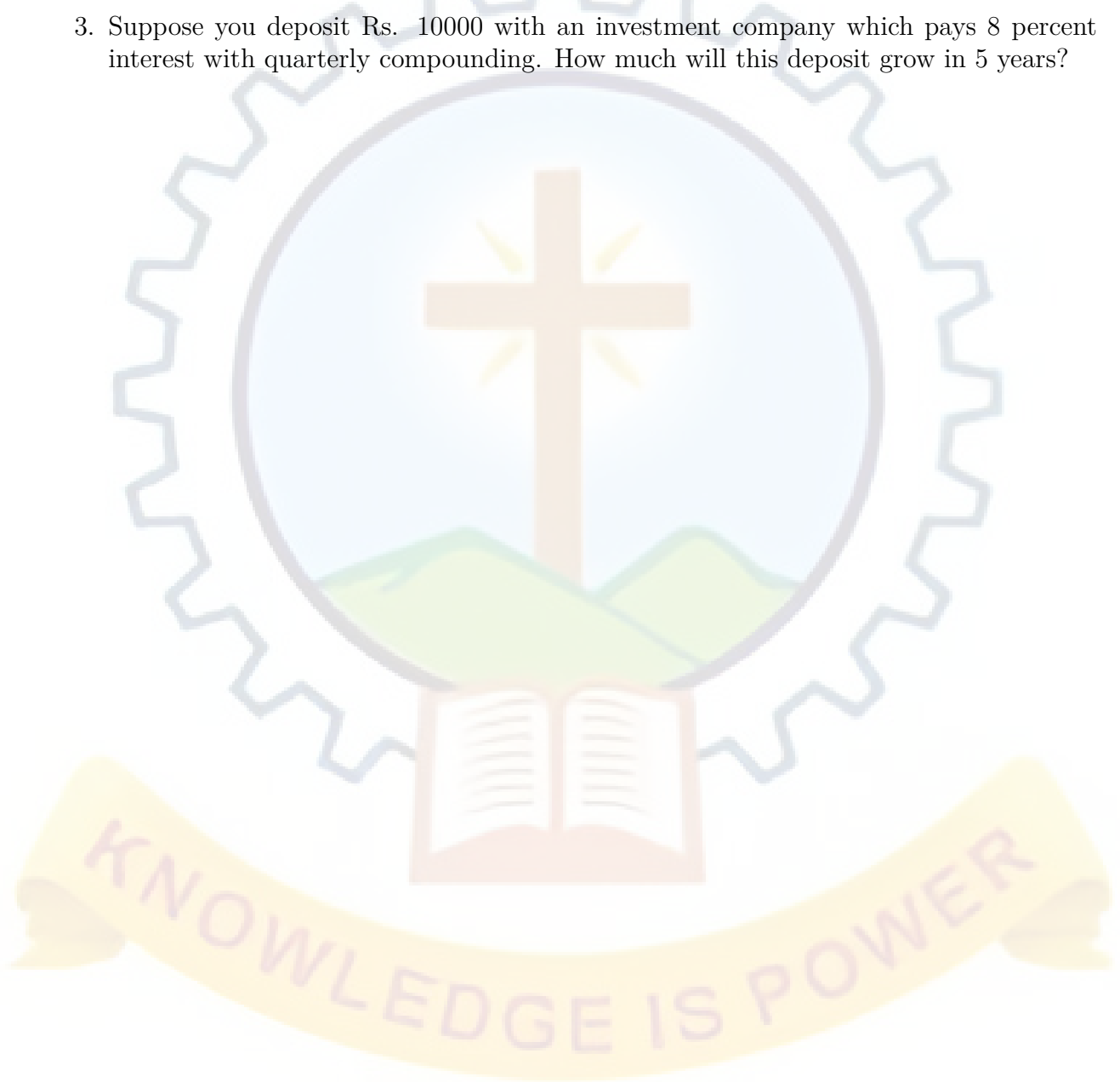
1. What are the life skills that a professional should have?
2. Explain how time management can help in work life balance.
3. What is the difference between intuition and lateral thinking? What is the value of L after you run the code below?

Course Outcome 3 (CO 3):

1. How a person can grow as a leader in an organization?
2. Discuss the term "Crisis management".
3. What are the differences between a team and a group?

Course Outcome 4 (CO 4):

1. A finance company advertises that it will pay a lumpsum of Rs. 10000 at the end of 6 years to investors who deposit annually Rs. 1000. What interest rate is implicit in this offer?
2. How much should be deposited at the beginning of each year for 10 years in order to provide a sum of Rs. 50000 at the end of 10 years?
3. Suppose you deposit Rs. 10000 with an investment company which pays 8 percent interest with quarterly compounding. How much will this deposit grow in 5 years?



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIRST SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2024

Course Code: B24MC1T01

Course Name: LIFE SKILLS

Max. Marks: 50

Duration: 2 hours

PART A

Answer all questions. Each question carries 5 marks.

1. Stress is the emotional or physical tension the body creates when presented with events or thoughts that cause worry, frustration, anger or nervousness. When stress exceeds the ability to cope, balance in the mind and body need to be restored. Discuss how stress management can act as an effective tool to accomplish this.
2. "The only limit to our realization of tomorrow is our doubts of today." – Franklin D. Roosevelt. Critically assess how cultivating positive thinking and maintaining a right attitude can transform professional challenges into opportunities for growth.
3. Discuss leadership styles that are effective for successful management of multicultural groups and teams.
4. Mr. Vinay plans to send his son for higher studies abroad after 10 years. He expects the cost of these studies to be Rs. 100000. How much should he save annually to have a sum of Rs. 100000 at the end of 10 years if the interest rate is 12 percent?

PART B

Read carefully the following case and answer the questions given below. Each question carries 6 marks.

1. Based on the case study given below, answer the following questions: It occurred on the night of 2–3 December 1984 at the Union Carbide India Limited (UCIL) pesticide plant in Bhopal, Madhya Pradesh. Over 500,000 people were exposed to Methyl Isocyanate (MIC) gas and other chemicals. A runaway reaction had occurred in a storage tank of Methyl Isocyanate (MIC), which was used to manufacture a pesticide. The valves of the tank had burst, and a cloud of poisonous gas had escaped. The winds carried it to nearby shanty towns and the populous city of Bhopal, where thousands of people either died in their sleep or woke and died while fleeing. Those who survived suffered from burning eyes and lungs. Local medical facilities were not equipped for the disaster, and over the next few weeks' thousands more died. The killer gas spread through the city, sending residents scurrying through the dark streets. No alarm ever sounded a warning, so that local people were not informed the situation, and no evacuation plan was prepared. When victims arrived at hospitals breathless and blind, doctors did not know how to treat them, as UCIL had not provided emergency information. Perhaps most importantly at the time of the tragedy, the staff did not realize the gravity of the situation and even took a break for tea after the leak had been noticed, thinking they would have plenty of time to fix it. The operator in the control room did not notify his supervisor when the temperature began to rise inside the tank and the entire situation remained unattended for at least an hour. The disaster raised some serious ethical issues. The pesticide factory was built in the midst of densely populated settlements. UCIL chose to store and produce MIC, one of the deadliest chemicals (permitted exposure levels in USA and Britain are 0.02 parts per million), in an area where nearly 120,000 people lived. The MIC plant was not designed to handle a runaway reaction. When the uncontrolled reaction started, MIC was flowing through the scrubber (meant to neutralize MIC emissions) at more than 200 times its designed capacity.
 - (a) Critique the communication strategy (or lack thereof) employed by UCIL during the disaster. How did the absence of timely warnings and information affect the outcome?
 - (b) Assess the ethical implications of UCIL's decision to build a pesticide plant in a densely populated area. How should corporate responsibility have been exercised in this context?
 - (c) As an engineer, comment on the drawback of the design which may have the reason for the tragedy.
 - (d) Evaluate the leadership displayed by UCIL's management during the Bhopal disaster. How did their response, or lack thereof, impact the outcome of the crisis?
 - (e) Reflect on the lessons learned from the Bhopal disaster. What key takeaways should industries and governments derive from this incident to enhance safety and prevent future catastrophes?

B24MC1T02	DESIGN THINKING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		1	1	0	2	P/F	2024

Preamble

This course gives students a comprehensive understanding of the iterative design process and its real-world applications. It covers the fundamentals of design thinking, including concept development, brainstorming, and creativity enhancement. Emphasizing customer needs identification and human-centered design principles, it explores product conceptualization and evaluation, along with prototyping techniques. Additionally, the course addresses ethical considerations and challenges within the design thinking process through diverse case studies. By the end of the course, students will gain practical insights into design thinking methodologies, preparing them to effectively tackle complex design challenges.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Exhibit/show a thorough understanding of the fundamental principles of the design thinking methodology. (Cognitive Knowledge Level: Understand)))
CO 2	Utilize diverse techniques effectively to generate creative concepts, adopting innovation and ideation. (Cognitive Knowledge Level: Apply))
CO 3	Demonstrate expertise in ideating prototypes, models, and proof-of-concept iterations. (Cognitive Knowledge Level: Apply)
CO 4	Analyze real-world challenges and develop a practical design thinking framework suitable for their professional endeavors. (Cognitive Knowledge Level: Create)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1	3	1	1	1	1	2	1	1	1
CO 2	2	2	3	1	1	1	1	2	1	1	1
CO 3	2	2	3	1	1	1	1	2	1	1	1
CO 4	2	2	3	1	1	1	1	2	1	1	1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test (%Marks)	Case Study Presentation (Marks) (%Marks)	
Remember	25		20
Understand	25		20
Apply	25		20
Analyse	25		20
Evaluate			
Create		100	20

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	50	50	2 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (1 number)	25 marks
Self Study (Case Study Presentation)	15 marks

End Semester Examination Pattern There will be two parts: Part A and Part B. Part A contains 4 questions carrying 5 marks each. Part B contains 2 questions from each module out of which 1 is to be answered and can have maximum 2 sub- divisions. Questions from Module 1 & 2 carry 8 marks each and Module 3 & 4 carry 7 marks.

SYLLABUS

MODULE 1 (5 hours)

Design Thinking Approach:

Introduction to Design Thinking; Iterative Design Thinking Process Stages: Empathize, Define, Ideate, Prototype and Test; The double-diamond Model of design by British Design Council.

Developing concepts:

Steps to develop concepts from functions; Brainstorming: Mechanism of brainstorming, Ideation; Creativity: How to increase level of creativity.

MODULE 2 (6 hours)

Design Process: Requirements: Identifying customer needs and requirements, market analysis, defining goals; Product concepts: establishing functions, task specifications.

Solution Concept: conceptualization, evaluating alternatives; embodiment design; Analysis and optimization; experiment; marketing. Human-centred design process.

MODULE 3 (6 hours)

Concepts Evaluation: Evaluating conceptual alternatives: Pugh's Evaluation matrix, decision matrix with examples, QFD and house of quality.

Prototyping: Prototypes, Models and Proofs of concepts; What is Prototype? Why Prototype? Building models and prototypes, Rapid Prototyping; Lean startup method for prototype development; Testing prototypes and models and proving concepts.

MODULE 4 (7 hours)

Ethics in Design: Understanding obligations, code of ethics, familiarity with several code of ethics such as ASCE, ASME, IEEE, VDI etc. code of ethics and moral frameworks.

Challenges in Design Thinking: Design thinking case studies detailing the various aspects detailed above are to be discussed. The case studies are suggested to be from the below listed areas but not to be limited to: Consumer package goods; Education; Financial Services; Health care; Journalism; Non-Profit organizations; Retail; Technology; Transportation sector; Self-improvement.

Text Books

1. Yousef Haik & Tamer M Shahin, "Engineering design process", Course Technology, 2010.
2. Clive L Dym, Patrick Little & Elizabeth J Orwin, "Engineering Design-A Project based Introduction", Wiley, 2014.

3. Don Norman, "The Design of Everyday Things", Basic Books; 2nd edition, 2013.
4. Christian Mueller-Roterberg, "Handbook of Design Thinking: Tips and Tools for how to design thinking", 2018.

Reference Books

1. Daniel Kahneman, "Thinking Fast and Slow", Farrar, Straus & Giroux, 201.
2. Rod Judkins, "The art of Creative Thinking", Penguin Publishing Group, 2016.
3. Donella H Meadows, "Thinking in Systems", Chelsea Green Publishing, 2008.
4. Tim Brown, "Change by Design", HarperCollins, 2019.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		24 Hours
Module 1		5 hours
1.1	Design Thinking Approach: Introduction to Design Thinking; Iterative Design Thinking Process Stages: Empathize, Define, Ideate, Prototype and Test	1
1.2	The double-diamond Model of design by British Design Council	1
1.3	Developing concepts: Steps to develop concepts from functions	1
1.4	Brainstorming: Mechanism of brainstorming, Ideation	1
1.5	Creativity: How to increase level of creativity	1
Module 2		6 hours
2.1	Design Process: Requirements: Identifying customer needs and requirements, market analysis, defining goals	1
2.2	Product concepts: establishing functions, task specifications	1
2.3	Time management: Prioritizing tasks, setting realistic goals and managing time effectively, work life balance.	1

2.4	Solution Concept: conceptualization, evaluating alternatives	1
2.5	Embodiment design; Analysis and optimization; experiment; marketing	1
2.6	Human-centred design process	1
Module 3		6 hours
3.1	Concepts Evaluation: Evaluating conceptual alternatives: Pugh's Evaluation matrix, decision matrix with examples.	1
3.2	Prototypes, Models and Proofs of concepts	1
3.3	What is a Prototype? Why Prototype? Building models and prototypes, Rapid Prototyping	1
3.4	Lean startup method for prototype development; Testing prototypes and models and proving concepts	1
3.5	QFD and House of Quality	1
Module 4		7 hours
4.1	Ethics in Design: Understanding obligations, code of ethics, familiarity with several code of ethics such as ASCE, IEEE, VDI etc. code of ethics and moral frameworks	1
4.2	Challenges in Design thinking	1
4.3	Design thinking case studies detailing the various aspects	5

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. Describe design thinking and list the different stages in a design thinking process.
2. Illustrate the double-diamond Model of design.
3. Describe how to develop concepts from functions and Mechanism of brainstorming.
4. How to increase the level of creativity and the process of forming ideas from conception to implementation?

Course Outcome 2 (CO 2):

1. How to narrow down to the best design considering the customer needs and requirements, market analysis and defining goals?

2. Illustrate the process of product concepts, forming ideas and embodiment design.
3. Explain the Human-centred design process.

Course Outcome 3 (CO 3):

1. Describe the concept evaluation using Pugh's Evaluation matrix and decision matrix with examples.
2. Explain the ideation of prototypes, models, and proofs of concept.
3. Illustrate the concept of Rapid Prototyping, the Lean startup method for prototype development and testing of prototypes.

Course Outcome 4 (CO 4):

1. Discuss as an engineer how ethics play a decisive role in design.
2. Analyze the Challenges in Design thinking.
3. Illustrate the development of a product by passing through the different stages of the design thinking Process.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

MAR ATHANASIUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM

FIRST SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2024

Course Code: B24MC1T02

Course Name: DESIGN THINKING

Max. Marks: 50

Duration: 2 hours

PART A

Answer all questions. Each question carries 5 marks.

1. Demonstrate the basic concept of brainstorming and the rules developed for brainstorming session.
2. Briefly explain what is product and solution concepts in design process.
3. Distinguish between prototypes and models.
4. Explain the importance of ethics in design.

PART B

Answer any one question from each module.

11. (a) What do you mean by design thinking and why it is needed. How does the design thinking approach help engineers. 8

OR

12. (a) Summarize different stages of design thinking process using appropriate examples.. 8
13. (a) Illustrate different phases of extensive prescriptive model of design process. 8

OR

14. (a) Identify the customer requirements with the help of refrigerator as example, 8
15. (a) How concepts evaluation can be done using Pugh's evaluation matrix. Compare Pugh's evaluation matrix with the decision matrix. 7

OR

16. (a) List the different methods in which the prototype of a product can be generated and tested. 7
17. (a) Design a device/machine that will crush aluminum cans. The device must be fully automatic. The device should switch on automatically, crush the can automatically, eject the crushed can automatically and switch off automatically. 7

OR

18. (a) Design a new shopping cart that can be used primarily in grocery stores. The shopping cart should solve the common problems in the available carts. There is a tendency to conserve parking space by not designating a return cart area. Leaving cart in the parking lots may lead to serious accidents and car damage. Many customers do not fill their carts when shopping; however, they do not like to carry baskets. Other customers like to sort products as they shop. 7

B24MC1L01	YOGA AND SPORTS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	1	1	1	P/F	2024

Preamble

This course enables the learners to understand how to attain physical fitness, mental well-being, and holistic growth through the combined benefits of yoga and sports. The topics covered in this course are Yoga & Lifestyle Physical fitness, wellness and exercise programmes, First aid and Postures & nutrition. This course helps the students to develop appreciation of physical activity as a lifetime pursuit and a means to better health.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Demonstrate the need of physical activities and Yoga for the strength, flexibility, and relaxation of mind and body. (Cognitive Knowledge Level: Apply)
CO 2	Use scientific principles of exercise and training in daily routine. (Cognitive Knowledge Level: Apply)
CO 3	Apply first aid promptly and appropriately whenever and wherever the need arises. (Cognitive Knowledge Level: Apply)
CO 4	Understand the importance of postures and nutrition (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1								2	3	2	2
CO 2								3	3	2	2
CO 3						2		3	3	3	2
CO 4								3	3	2	2

Mark Distribution

Total Marks	CIE Marks
50	50

Continuous Internal Evaluation Pattern

Attendance	10 marks
Regular Assessment	35 marks
Self Study	5 marks

Marks for the regular assessment can be based on the CO questions given at the end.

SYLLABUS

MODULE 1 (6 hours)

Yoga & Lifestyle:

Meaning and importance of Yoga. Introduction-Asanas: Pranayama, Meditation and Yogic Kriyas. Yoga for concentration and related Asanas (Sukhasana; Tadasana; Padmasana and Shashankasana). Relaxation Techniques for improving concentration-Yog-nidra. Asanas as preventive measure. Hypertension: Tadasana, Vajrasana, Pawanuktasana, Ardha Chakrasana, Bhujangasana, Shavasana. Obesity: Procedure, Benefits and contraindications for Vajrasana, Hastasana, Trikonasana, Ardha Matsyendrasana. Back pain: Tadasana, Ardha Matsyendrasana, Vakrasana, Shalabhasana, Bhujangasana.

MODULE 2 (6 hours)

Physical fitness and exercise:

Meaning and importance of physical fitness and wellness. Components of physical fitness and health related fitness. Exercise for improving speed, strength, endurance, and flexibility and coordinative abilities. Exercises to prevent back pain, tennis elbow, shoulder injury and knee pain, Neck pain. Fitness test battery for speed, strength, endurance, flexibility. Importance of weight training. Warming up and cooling down. How to deal with every day stress.

MODULE 3 (6 hours)

First aid:

First aid and principles of first aid. First aid measure for the following: Bleeding through Nose, Snakebite, Dog Bite, Electric Shock, Burns and Drowning. Common injuries and

their management: Wounds, Cuts, Sprain, Fracture and Dislocation. Cardio Pulmonary Resuscitation (CPR). How to prevent muscle cramps and its management. How to carry an injured person.

MODULE 4 (6 hours)

Postures and nutrition:

Posture and its importance. Common Postural Deformities- Knock Knee, Flat Foot, Round Shoulders, Lordosis, Kyphosis, Bow Legs and Scoliosis. Corrective Measures for Postural Deformities. Balanced diet, malnutrition and Deficiency diseases. Hydration.

Text Books

1. Modern Trends and Physical Education by Prof. Ajmer Singh.
2. Light on Yoga by B.K.S. Iyengar.
3. Health and Physical Education- NCERT (11th and 12th Classes)

Reference Books

1. Physiological aspects of sports training and performance by Jay Hoffman.
2. Periodization theory and methodology of training by Tudor O Bompa and G Grigory Haff.
3. Essential of strength training and conditioning by Thomas Baechle E R, Roger W Earle.
4. A practice guide to emergency first aid, safety injuries, illnesses by Montreal.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		24 Hours
Module 1: Yoga & Lifestyle		6 hours
1.1	Meaning and importance of Yoga. Introduction-Asanas, Pranayama, Meditation and Yogic Kriyas. Yoga for concentration and related Asanas (Sukhasana; Tadasana; Padmasana and Shashankasana) Relaxation Techniques for improving concentration-Yog-nidra. Asanas as preventive measures.	2

1.2	Hypertension: Tadasana, Vajrasana, Pavan Muktasana, Ardha Chakrasana, Bhujangasana, Sharasana.	1
1.3	Obesity: Procedure, Benefits and contraindications for Vajrasana, Hastasana, Trikonasana, Ardh Matsyendrasana.	1
1.4	Back pain: Tadasana, Ardh Matsyendrasana, Vakrasana, Shalabhasana, Bhujangasana	2
Module 2: Physical fitness and exercise		6 hours
2.1	Meaning and importance of physical fitness and wellness, Components of physical fitness and health-related fitness	1
2.2	Exercise for improving speed, strength, endurance, and flexibility and co ordinative abilities	1
2.3	Exercises to prevent back pain, shoulder injury and knee pain.	2
2.4	Fitness test battery for speed, strength, endurance, flexibility.	1
2.5	Importance of weight training, Warming up and cooling down.	1
Module 3: First aid		6 hours
3.1	First aid and principles of first aid.First aid measure for the following: Bleeding through Nose, Snakebite, Dog Bite, Electric Shock, Burns and Drowning.	2
3.2	Common injuries and their management: Wounds, Cuts, Sprain, Fracture and Dislocation	2
3.3	Cardio pulmonary resuscitation (CPR).	1
3.4	How to prevent muscle cramps and its management.How to carry an injured person	1
Module 4: Postures and nutrition		6 hours
4.1	Posture and its importance.Common Postural Deformities-Knock Knee, Flat Foot, Round Shoulders.	2
4.2	Lordosis, Kyphosis, Bow Legs and Scoliosis.Corrective Measures for Postural Deformities.	2
4.3	Balanced diet, malnutrition and deficiency disease, Hydration.	2

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. Demonstrate yoga asanas for life style problems
2. Create a PPT presentation on various yoga asanas
3. Group Activity - Group discussion about the need and benefits of physical activities and Yoga for the strength, flexibility, and relaxation of mind and body.

Course Outcome 2 (CO 2):

1. Analyze the exercise activities of at least five famous personalities and give a PPT presentation about how each one of them uses physiological principles related to exercise and training in daily routine.
2. Conduct a survey on how the following categories of people follow physiological principles related to exercise and training in daily routine.
 - (a) Sports person
 - (b) Working woman
 - (c) Students
 - (d) Ladies in the age group of 25-35, 35-45, 45- 55, 55-65, above 65
 - (e) Gents in the age group of 25-35, 35-45, 45- 55, 55-65, above 65

Course Outcome 3 (CO 3):

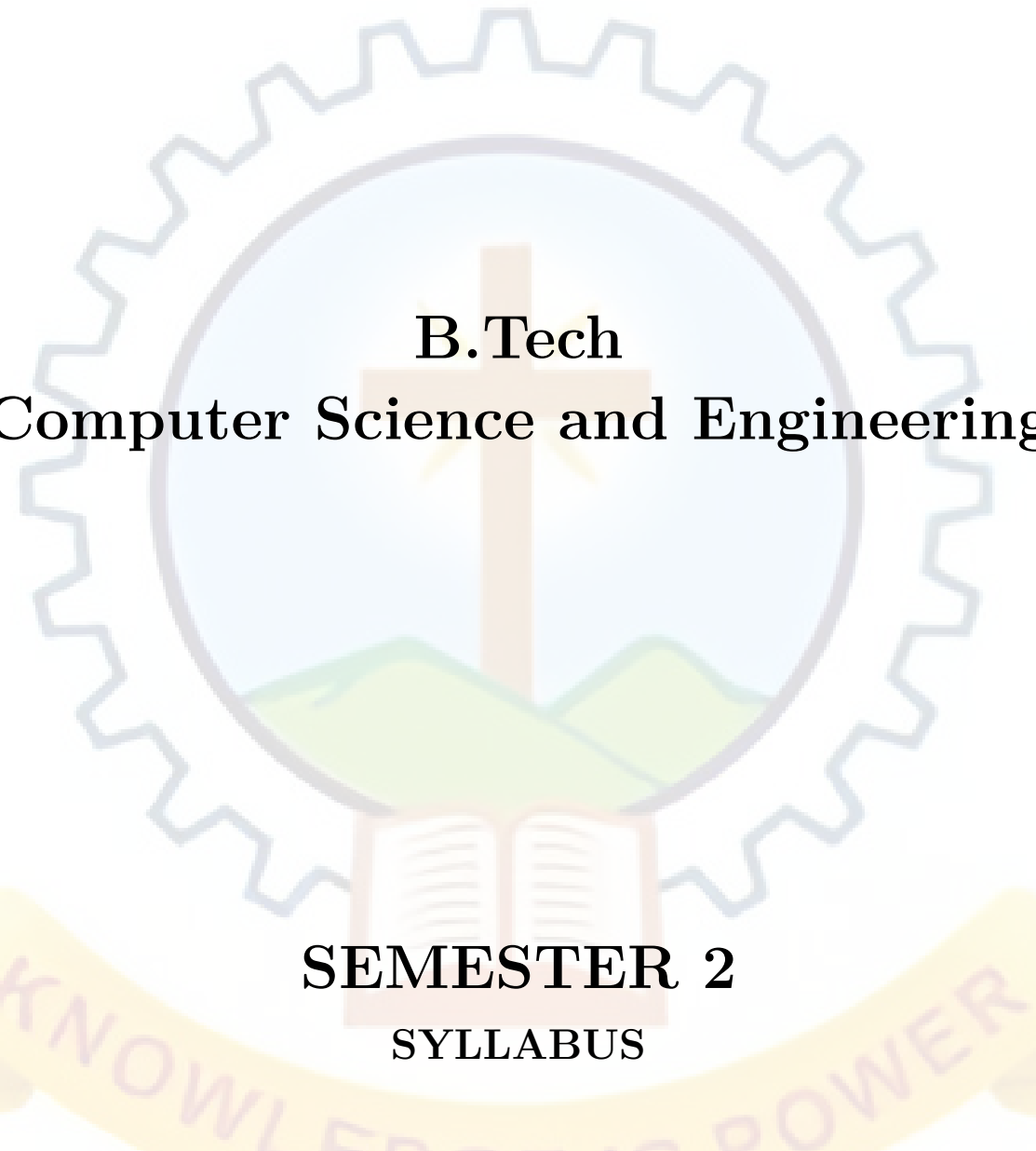
1. With a role play, illustrate various first aid activities that can be followed in various situations in life. In each illustration, try to give emphasis on dos and don'ts to be followed in each situation.

Course Outcome 4 (CO 4):

1. Observe at least 10 students in your class and identify Common Postural deformities each one of them have. Also identify good postures they follow. Have a discussion with each one of them to identify whether they have already recognized it or not. Make a report about it.

MAR ATHANASIOUS COLLEGE OF ENGINEERING

Government Aided, Autonomous Institution
Kothamangalam, Kerala, India

The logo of Mar Athanasius College of Engineering is a circular emblem. It features a large gear-like outer border. Inside the circle, there is a light blue sky with a white cross in the center. Below the cross are green hills. At the bottom of the circle is an open book. A yellow banner with the text 'KNOWLEDGE IS POWER' in purple capital letters curves around the bottom of the emblem.

B.Tech Computer Science and Engineering

SEMESTER 2 SYLLABUS

SEMESTER 2

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24MA1T02	ORDINARY DIFFERENTIAL EQUATIONS AND TRANSFORMS	3-1-0-5	4	4
B	B24ES1T03A	COMPUTER AIDED ENGINEERING GRAPHICS (A)	2-0-2-4	4	3
C	B24CS1T01	LOGIC SYSTEM DESIGN	3-1-0-5	4	4
D	B24CS1T02	INDUSTRIAL PROGRAMMING	2-1-0-4	3	3
E	B24CS1T03	OBJECT ORIENTED PROGRAMMING	3-1-0-5	4	4
G	B24CS1L01	INDUSTRIAL PROGRAMMING LAB	0-0-3-2	3	2
H	B24CS1L02	OBJECT ORIENTED PROGRAMMING LAB	0-0-3-2	3	2
I	B24MC1T03	PROFESSIONAL COMMUNICATION & ETHICS	2-0-1-4	3	P/F
J	B24MC1L02	IDEA LAB	0-0-2-2	2	P/F
TOTAL				30	22

B24MA1T02	ORDINARY DIFFERENTIAL EQUATIONS AND TRANSFORMS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble:

This course introduces the concepts and applications of differential equations, sequence and series including power series and basic transforms such as Laplace and Fourier transforms. The objective of this course is to familiarize the prospective engineers with some advanced concepts and methods in Mathematics which include differential equations, sequence, series and transforms. The topics treated in this course have applications in all branches of engineering.

Prerequisites: Nil

Course Outcomes:

After the completion of the course the student will be able to:

CO 1	Solve homogeneous and non-homogeneous ordinary differential equations using appropriate analytical methods. (Cognitive Knowledge Level: Apply)
CO 2	Determine the convergence and divergence of sequences and series using standard convergence tests. (Cognitive Knowledge Level: Apply)
CO 3	Apply Taylor series, Fourier series, and half-range expansions to represent functions. (Cognitive Knowledge Level: Apply)
CO 4	Determine the Fourier transforms of functions and apply them to solve problems arising in engineering. (Cognitive Knowledge Level: Apply)
CO 5	Apply Laplace transforms to model and solve engineering systems governed by ordinary differential equations. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	1	1									
CO 2	1	1				1					
CO 3	1	1									
CO 4	1	1			1						
CO 5	1	1		1	1	1	1				1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (% Marks)	Test 2 (% Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (Ordinary Differential Equations)

(Text 2: Relevant topics from sections 2.1, 2.2, 2.5,2.6, 2.7, 2.10, 3.1, 3.2, 3.3)

Homogenous linear differential equation of second order, superposition principle, general solution, homogenous linear ODEs with constant coefficients-general solution. Solution of Euler-Cauchy equations (second order only). Existence and uniqueness (without proof). Non homogenous linear ODEs-general solution, solution by the method of undetermined coefficients(for the right hand side of the form $x^n, e^{kx}, \sin ax, \cos ax$ and their linear combinations) , methods of variation of parameters. Solution of higher order equations-homogeneous and non-homogeneous with constant coefficients using method of undetermined coefficients.

MODULE 2 (Sequences and Series)

(Text 1: Relevant topics from sections 9.1, 9.3, 9.4, 9.5, 9.6)

Convergence of sequences and series, convergence of geometric series and p-series (without proof), tests of convergence (comparison, limit comparison, ratio and root tests without proof); Alternating series and Leibnitz test, absolute and conditional convergence.

MODULE 3 (Fourier Series)

(Text 1: Relevant topics from sections 9.8, 9.9. Text 2: Relevant topics from sections 11.1, 11.2, 11.6)

Taylor series (without proof, assuming the possibility of power series expansion in appropriate domains), Binomial series and series representation of exponential, trigonometric, logarithmic functions (without proofs of convergence); Fourier series, Euler formula, Convergence of Fourier series (without proof), half range sine and cosine series.

MODULE 4 (Fourier Transforms)

(Text 2: Relevant topics from sections 11.7, 11.8, 11.9)

Fourier integral representation, Fourier sine and cosine integrals. Fourier sine and cosine transforms, inverse sine and cosine transform. Fourier transform and inverse Fourier transform, basic properties. The Fourier transform of derivatives. Convolution theorem (without proof).

MODULE 5 (Laplace Transforms)

(Text 2: Relevant topics from sections 6.1, 6.2 ,6.3, 6.4, 6.5)

Laplace Transform and its inverse, Existence theorem (without proof), linearity, Laplace transform of basic functions, first shifting theorem, Laplace transform of derivatives and integrals, solution of differential equations using Laplace transform, Unit step function,

Second shifting theorem. Dirac delta function and its Laplace transform, Solution of ordinary differential equation involving unit step function and Dirac delta functions. Convolution theorem (without proof) and its application to finding inverse Laplace transform of products of functions.

Text Books

1. H. Anton, I. Biven, S. Davis, "Calculus", Wiley, 10th edition, 2015.
2. Erwin Kreyszig, "Advanced Engineering Mathematics", 10th edition, John Wiley & Sons, 2016.

Reference Books

3. J. Stewart, "Essential Calculus", Cengage, 2nd edition, 2017.
4. G.B. Thomas and R.L. Finney, "Calculus and Analytic geometry", 9th Edition, Pearson, Reprint, 2002.
5. Peter O Neil, "Advanced Engineering Mathematics", 7th Edition, Thomson, 2007.
6. Louis C Barret, C Ray Wylie, "Advanced Engineering Mathematics", Tata McGraw Hill, 6th edition, 2003.
7. Veerarajan T, "Engineering Mathematics for first year", Tata McGraw - Hill, 2008.
8. B.S. Grewal, "Higher Engineering Mathematics", Khanna Publishers, 43 Edition, 2015.
9. Ronald N. Bracewell, "The Fourier Transform and its Applications", McGraw – Hill International Editions, 2000.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Ordinary Differential Equations		9 hours
1.1	Homogenous linear equation of second order, Superposition principle, general solution.	1
1.2	Homogenous linear ODEs of second order with constant coefficients.	2
1.3	Second order Euler-Cauchy equation.	1

1.4	Non homogenous linear differential equations of second order with constant coefficient-solution by undetermined coefficients, variation of parameters.	3
1.5	Higher order equations with constant coefficients.	2
Module 2: Sequences and Series		9 hours
2.1	Convergence of sequences and series, geometric and p-series.	2
2.2	Test of convergence (comparison, ratio and root).	4
2.3	Alternating series and Leibnitz test, absolute and conditional convergence	3
Module 3: Fourier series		9 hours
3.1	Taylor series, Binomial series and series representation of exponential, trigonometric, logarithmic functions.	3
3.2	Fourier series, Euler formulas, Convergence of Fourier series (Dirichlet's conditions)	3
3.3	Half range sine and cosine series.	3
Module 4: Fourier Transforms		9 hours
4.1	Fourier integral representation.	1
4.2	Fourier Cosine and Sine integrals and transforms.	2
4.3	Complex Fourier integral representation, Fourier transform and its inverse transforms, basic properties.	3
4.4	Fourier transform of derivatives, Convolution theorem	3
Module 5: Laplace Transforms		9 hours
5.1	Laplace Transform , inverse Transform, Linearity, First shifting theorem, transform of basic functions.	2
5.2	Transform of derivatives and integrals.	1
5.3	Solution of Differential equations, Initial value problems by Laplace transform method.	2
5.4	Unit step function - Second shifting theorem.	1
5.5	Dirac Delta function and solution of ODE involving Dirac delta function.	2
5.6	Convolution and related problems.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1): Solve homogeneous and nonhomogeneous linear equation with constant coefficients.

1. Find the general solution to $2x^2 \frac{d^2y}{dx^2} + x \frac{dy}{dx} - 3y = 0$ given that $y_1(x) = \frac{1}{x}$ is a solution.
2. Solve the initial value problem $x^2 y'' - 3xy' + 4y = 0$ given that $y(1) = \pi, y'(1) = 4\pi$
3. By the method of undetermined coefficients, solve $y'' - 2y' + y = e^x \cos 2x$

Course Outcome 2 (CO 2): Perform various tests to determine whether a given series is convergent, absolutely convergent or conditionally convergent.

1. Find the sum of the series $\sum_{n=1}^{\infty} \frac{1}{9n^2+3n-2}$, if it is convergent.
2. Examine the convergence of $\sum_{n=1}^{\infty} \left(\frac{n}{n+1} \right)^{n^2}$
3. Determine whether the series $\sum_{n=1}^{\infty} \frac{(-1)^n n^4}{4^n}$ is absolutely convergent.

Course Outcome 3 (CO 3): Determine the power series expansion of a given function.

1. Find the Taylor's series representation of $f(x) = \sin \pi x$ about $x = 1$
2. Determine the binomial series representation of $\frac{1}{\sqrt{(2+x)^3}}$
3. Find the Fourier series of the periodic function $f(x)$ of period 2, where
$$f(x) = \begin{cases} -1 & -1 \leq x \leq 0 \\ 2x & 0 \leq x \leq 1 \end{cases}$$
 and deduce that $1 + \frac{1}{3^2} + \frac{1}{5^2} + \dots = \frac{\pi^2}{8}$.

Course Outcome 4 (CO 4): Determine the Fourier transforms of functions and apply them to solve problems arising in engineering .

1. Find the Fourier integral representation of function defined by $f(x) = e^{-x}$ for $x > 0$ and $f(x) = 0$ for $x < 0$.
2. What are the conditions for the existence of Fourier Transform of a function $f(x)$?
3. Find the Fourier Transform of $f(x) = x$ for $|x| \leq 1$ and $f(x) = 0$ otherwise.

Course Outcome 5 (CO 5): Compute Laplace transform and apply them to solve ODEs arising in engineering.

1. What is the inverse Laplace Transform of $\frac{3s+2}{(s-1)(s^2+2s+5)}$
2. Find Laplace Transform of (i) $e^{-t}\sin^2 t$ (ii) $\delta(t-a)$
3. Solve the differential equation $y'' + 4y = f(t)$, $y(0) = 1$, $y'(0) = 0$ where $f(t) = \begin{cases} 0 & \text{if } 0 \leq t \leq 4 \\ 3 & \text{if } t \geq \pi \end{cases}$



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM

SECOND SEMESTER B.TECH DEGREE EXAMINATION, JUNE 2025

Course Code: B24MA1T02

Course Name: **ORDINARY DIFFERENTIAL EQUATIONS AND
TRANSFORMS** Common to all branches

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Check whether $x, \ln x$ are linearly independent or not.
2. Solve $y''' + 9y' = 0$.
3. Find the rational number represented by the repeating decimal 5.373737...
4. Examine the convergence of $\sum_{k=1}^{\infty} \frac{1}{k!}$
5. Find the binomial series for $f(x) = (1+x)^{\frac{1}{3}}$ upto third degree term.
6. Obtain the half range sine series expansion of $f(x) = \pi x - x^2$ in $(0, \pi)$.
7. Find the cosine integral representation of the function $f(x) = \begin{cases} 1 & ; 0 < x < 1 \\ 0 & ; x > 1 \end{cases}$
8. Find the Fourier cosine transform of e^{-x} , $x > 0$.
9. Find the Laplace transform of $\sin^2 2t$.
10. Find $L^{-1} \left\{ \frac{1}{(s-1)(s-2)} \right\}$.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Solve the initial value problem $y'' + 9y = 0, y(0) = 0.2, y'(0) = -1.5$. 7
 (b) By the method of variation of parameters solve $y'' + 4y = \tan 2x$. 7

OR

12. (a) By the method of undetermined coefficients solve $y'' + 2y' + 4y = 3e^{-x}$. 7
 (b) Solve $x^2y'' + xy' + 9y = 0, y(1) = 0, y'(1) = 2.5$. 7
13. (a) Test the convergence of (i) $\sum_{k=1}^{\infty} \frac{3k^3 - 2k^2 + 4}{k^7 - k^3 + 2}$ (ii) $\sum_{k=1}^{\infty} \frac{k^k}{k!}$. 7
 (b) Check the convergence of the series $1 + \frac{1.3}{3!} + \frac{1.3.5}{5!} + \frac{1.3.5.7}{7!} + \dots$ 7

OR

14. (a) Determine whether the series $\sum_{k=1}^{\infty} \frac{1}{\sqrt{k+1}}$ is absolutely convergent or conditionally convergent. 7
 (b) Test the convergence of (i) $\sum_{k=1}^{\infty} \frac{k!}{3!(k-1)!3^k}$ (ii) $\sum_{k=1}^{\infty} \left(\frac{4k-5}{2k+1}\right)^k$ 7
15. (a) Expand into a Fourier series, $f(x) = e^{-x}, 0 < x < 2\pi$. 7
 (b) Obtain the half range Fourier sine series of $f(x) = \begin{cases} x & , 0 < x < \frac{\pi}{2} \\ \pi - x & , \frac{\pi}{2} < x < \pi \end{cases}$ 7

OR

16. (a) Find the Fourier series expansion of $f(x) = x^2$ in the interval $-\pi < x < \pi$.
 Hence show that $1 - \frac{1}{2^2} + \frac{1}{3^2} - \frac{1}{4^2} + \dots = \frac{\pi^2}{12}$. 7
 (b) Find the half range cosine series for $f(x) = (x-1)^2$ in $0 \leq x \leq 1$. 7
17. (a) Find the Fourier transform of $f(x) = \begin{cases} 1 & \text{if } |x| < 1 \\ 0 & \text{otherwise} \end{cases}$ 7
 (b) Find the Fourier sine integral of $f(x) = \begin{cases} \sin x & , 0 \leq x \leq \pi \\ 0 & , x > \pi \end{cases}$ 7

OR

18. (a) Using Fourier integral representation show that $\int_0^{\infty} \frac{\cos wx}{1+w^2} dw = \frac{\pi}{2} e^{-x}, x > 0$. 7
 (b) Find the Fourier sine transform of $f(x) = \begin{cases} k & , 0 < x < a \\ 0 & , x > a \end{cases}$ 7
19. (a) Find the Laplace transform of (i) $t \sin 2t$ (ii) $e^{-t} \sin 3t \cos 2t$ 7
 (b) Using convolution theorem find $L^{-1} \left\{ \frac{1}{s(s^2+4)} \right\}$ 7

OR

20. (a) Find $L^{-1} \left\{ \frac{4s+5}{(s+2)(s-1)^2} \right\}$ 7
 (b) Use Laplace transform to solve $y'' + 2y' + 2y = 0, y(0) = y'(0) = 1$. 7

B24ES1T03A	COMPUTER AIDED ENGINEERING GRAPHICS (A)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	0	2	4		2024

Preamble

This course aims to equip students with the skills for precise technical communication using global standards. Through this course, students learn to proficiently use CAD software and interpret engineering drawings accurately. Emphasis is placed on conveying design intent and specifications effectively. By mastering these skills, students develop a critical eye for detail and enhance their ability to communicate complex engineering concepts visually. Ultimately, the course prepares students to excel in the global engineering landscape by fostering proficiency in graphical communication and CAD expertise.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Draw the projection of points and lines located in different quadrants. (Cognitive Knowledge Level: Analyze)
CO 2	Prepare multi-view orthographic projections of objects by visualizing them in different positions. (Cognitive Knowledge Level: Apply)
CO 3	Draw sectional views and develop surfaces of a given object. (Cognitive Knowledge Level: Apply)
CO 4	Familiarize the tools and features of CAD software (Cognitive Knowledge Level: Understand)
CO 5	Prepare pictorial drawings using the principle of isometric projections and convert 3D views to orthographic views using CAD Software (Cognitive Knowledge Level: Analyze)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	2	2	2			1		1		1
CO 2	3	2	2	2			1		1		1
CO 3	3	2	2	2			1		2		1
CO 4	3	1	1	1	2		1		2		1
CO 5	3	2	2	2	2		1		2		1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember			
Understand	40	40	30
Apply	30	30	40
Analyse	30	30	30
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	100	50	2 hours

Continuous Internal Evaluation Pattern

Attendance	20 marks
Continuous Assessment Test (2 numbers)	40 marks
Assignment/Class work	30 marks
Self Study	10 marks

Continuous Assessment Test 1 will have 20 marks and will be from Part A. Test 2 will be from Part B and will also carry 20 marks. Regarding Assignments/Class work, 15 marks will be awarded for Part A and the remaining 25 marks should be based on class works/assignments from Part B (minimum 5 exercises).

End Semester Examination Pattern

ESE will have questions only from Part A and with a duration of 2-hours. The exam will be for 50 marks and will have to be drawn on A4 size answer booklets. The question paper shall contain two parts; Part I contains three questions, one question each from the three modules, each carrying 12 marks. Part II contains two questions (from any of the three modules) carrying 14 marks each. The student has to answer all the three questions from Part I and any one question from Part II.

SYLLABUS

PART A

MODULE 1 (11 hours)

Introduction:

Relevance of technical drawing in engineering field. Types of lines, Dimensioning, BIS code of practice for technical drawing.

Orthographic projection of Points and Lines:

Projection of points in different quadrants, Projection of straight lines inclined to one plane and inclined to both planes. Trace of line. Inclination of lines with reference planes True length of line inclined to both the reference planes.

MODULE 2 (10 hours) Orthographic projection of Solids:

Projection of Simple solids such as Triangular, Rectangle, Square, Pentagonal and Hexagonal Prisms, Pyramids, Cone and Cylinder. Projection of solids in simple position. Projection of solids with axis inclined to one of the reference planes and both reference planes.

MODULE 3 (10 hours) Sections of Solids:

Sections of Prisms, Pyramids, Cone, Cylinder with axis in vertical position and cut by different section planes. True shape of the sections.

Development of Surfaces:

Development of surfaces of the Prisms, Pyramids, Cone, Cylinder cut by different section planes.

PART B

MODULE 4 (6 hours)

Introduction to Computer Aided Drawing:

Role of CAD in design and development of new products, Advantages of CAD- Create a new drawing, Set model environment i.e., units, limits etc., Set interface settings e.g., snap, grid, ortho, Create and save an AutoCAD drawing template, Use zooming tools, Drawing commands as line, spline, circle, arc, rectangle, polygon, ellipse, Hatch a closed entity to represent sections, Erase & oops, Copy and Move objects, Rotate, Scale, Stretch Extend & Offset, Mirror and array, Apply Chamfers and Fillets, Edit polylines and spline, decurve, fit, thickness join & explode, Trim, break, explode, Create layers and assign properties as line weights, line types, colour, Modify status: On, Off, Freeze, Thaw, Lock, Unlock, Set layer current, Modify layer attributes, Text and Dimensions, Plotting, Extrusion.

MODULE 5 (8 hours)

Isometric Projection using CAD:

Isometric View and Projections of Prisms, Pyramids, Cone, Cylinder, Frustum of Pyramid, Frustum of Cone.

Conversion of Pictorial Views using CAD: Creating two-dimensional drawing from pictorial views.

Text Books

1. Bhatt, N.D., Engineering Drawing, Charotar Publishing House Pvt. Ltd.
2. John, K.C. Engineering Graphics, Prentice Hall India Publishers.
3. K.N. Anilkumar, Engineering Graphics, Adhyuth Narayan Publishers.
4. P. I. Varghese, Engineering Graphics, Tata McGraw Hill Education.

Reference Books

1. Agrawal, B. and Agrawal, C.M., Engineering Drawing, Tata McGraw Hill Publishers.
2. Duff, J.M. and Ross, W.A., Engineering Design and Visualisation, Cengage Learning.
3. Kulkarni, D.M., Rastogi, A.P. and Sarkar, A.K., Engineering Graphics with AutoCAD, PHI.
4. Luzaddff, W.J. and Duff, J.M., Fundamentals of Engineering Drawing, PHI.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction and Orthographic projection of Points and Lines		11 hours
1.1	Relevance of technical drawing in Engineering field. Types of lines, Dimensioning, BIS code of practice for technical drawing.	1
1.2	Concept of principle planes of projection, different quadrants, locating points on different quadrants	2

1.3	Projection of lines, inclined to one plane and Lines inclined to both planes.	4
1.4	Problems on lines using trapezoid method .	2
1.5	Line rotation method of solving, problems on line rotation method.	2
Module 2: Orthographic projection of Solids		10 hours
2.1	Introduction of different solids, Simple position plan and elevation of solids.	3
2.2	Problems on views of solids inclined to one plane.	2
2.3	Problems on views of solids inclined to both planes.	3
2.4	Practice problems on solids inclined to both planes.	2
Module 3: Sections of solids and development of surfaces		10 hours
3.1	Introduction to section planes. Principle of locating cutting points and finding true shape.	2
3.2	Problems on sections of different solids and Problems when the true shape is given.	3
3.3	Principle and development of simple solids.	2
3.4	Development of solids and sectioned solids.	3
Module 4: Introduction to Computer Aided Drawing		6 hours
4.1	Role of CAD in design and development of new products, Advantages of CAD	1
4.2	AutoCAD Fundamentals: Open, (and close) AutoCAD application, Create a new drawing, Set model environment ie units, limits etc, Set interface settings eg snap, grid, ortho, Create and save an AutoCAD drawing template.	1
4.3	Use zooming tools, Drawing commands as line, spline, circle, arc, rectangle, polygon, ellipse, Hatch a closed entity to represent sections, Erase & oops, Copy and Move objects, Rotate, Scale, Stretch Extend & Offset, Mirror and array, Apply Chamfers and Fillets, Edit polylines and spline, decurve, fit, thickness join & explode .	2
4.4	Trim, break, explode, Create layers and assign properties as line weights, line types, colour, Modify status: On, Off, Freeze, Thaw, Lock, Unlock, Set layer current, Modify layer attributes, Text and Dimensions, Plotting, Extrusion. .	2

Module 5: Isometric Projection using CAD		8 hours
5.1	Isometric View and Projections of Prisms, Pyramids, Cone, Cylinder.	3
5.2	Isometric View and Projections of Frustum of Pyramid, Frustum of Cone..	2
5.3	Creating two-dimensional drawing from pictorial views..	3

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. Locate points in different quadrants as per given conditions.
2. Problems on lines inclined to both planes.
3. Find True length, Inclinations and Traces of lines.

Course Outcome 2 (CO 2):

1. Draw orthographic views of solids and combination solids .
2. Draw views of solids inclined to any one reference plane.
3. Draw views of solids inclined to both reference planes.

Course Outcome 3 (CO 3):

1. Draw views of solids sectioned by a cutting plane.
2. Find location and inclination of cutting plane given true shape of the section.
3. Draw development of lateral surface of solids and also its sectioned views.

Course Outcome 4 (CO 4):

1. Draw the given figure including dimensions using 2D software.

Course Outcome 5 (CO 5):

1. Draw Isometric views/projections of solids/combination of solids using modelling software.
2. Create 2D model using modelling software from the given 3D figure or from real 3D objects.

MODEL QUESTION PAPER

QP CODE:

Pages: 1

Reg.No.:

Name:

**MAR ATHANASIUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SECOND SEMESTER B.TECH DEGREE EXAMINATION, JUNE 2025

Course Code: B24ES1T03A

Course Name: COMPUTER AIDED ENGINEERING GRAPHICS

Max. Marks: 50

Duration: 2 hours

Instructions: Retain construction lines. Show necessary dimensions.

PART I

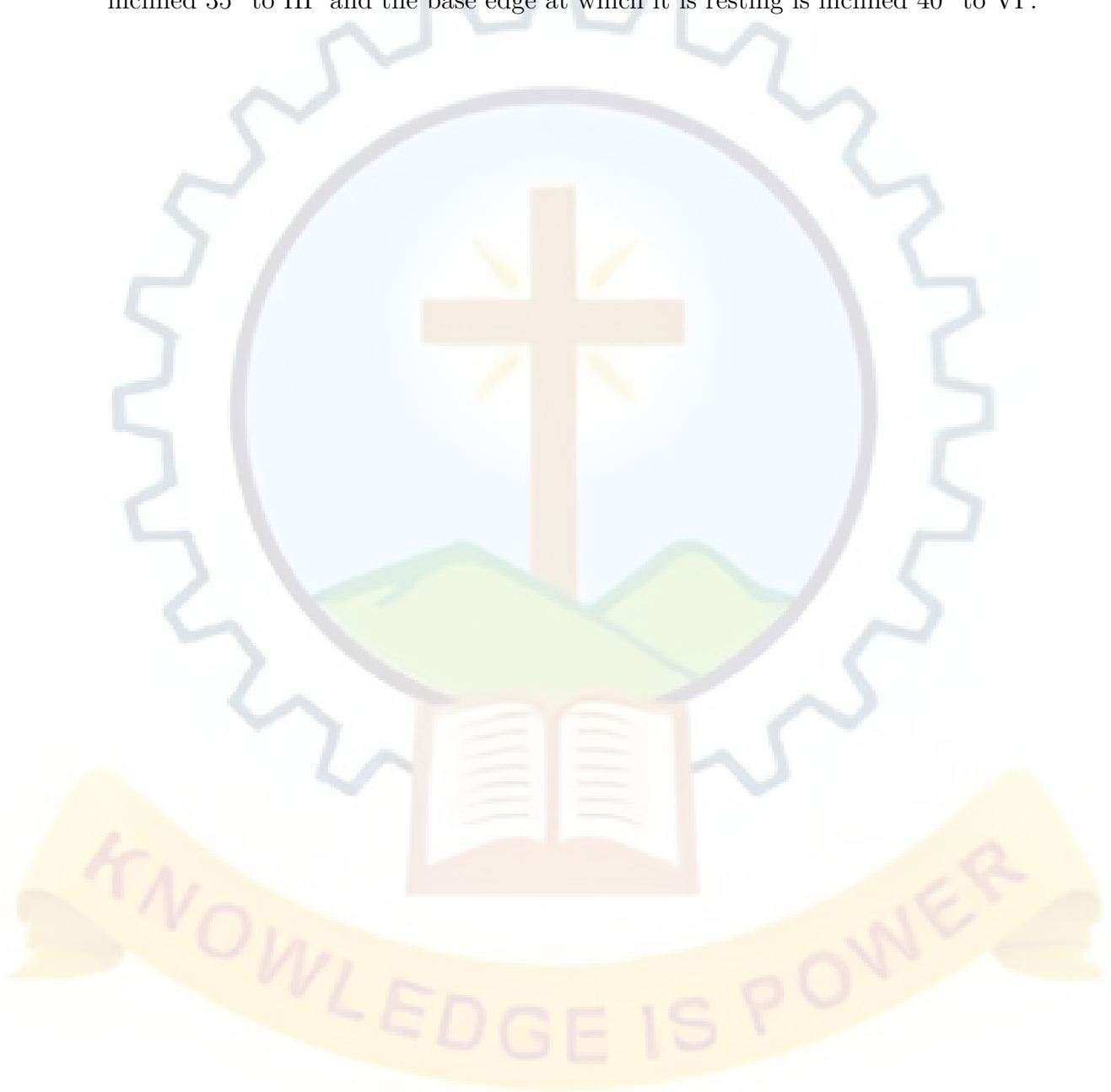
Answer all questions. Each question carries 12 marks.

1. The end point A of a line is 20mm above HP and 10mm in front of VP. The other end of the line is 50mm above HP and 15mm behind VP. The distance between the end projectors is 70mm. Draw the projections of the line. Find the true length and true inclinations of the line with the principal planes.
2. A pentagonal pyramid of base side 25mm and height 40mm, is resting on the ground on one of its triangular faces. The base edge of that face is inclined 30° to VP. Draw the projections of the solid.
3. Draw the development of a pentagonal pyramid of base side 30mm and height 50mm. A string is wound from a corner of the base round the pyramid and back to the same point through the shortest distance. Show the position of the string in the elevation and plan.

PART II

Answer any one full question. Each question carries 14 marks.

4. A triangular prism of base side 40mm and height 70mm is resting with its base on the ground and having an edge of the base perpendicular to VP. Section the solid such that the true shape of the section is a trapezium of parallel sides 30mm and 10mm. Draw the projections showing the true shape. Find the inclination of the cutting plane with the ground plane.
5. A hexagonal prism of base edge 25 mm and height 60 mm is resting on one of its base edges on HP. Draw its projection if the rectangular face carrying that base edge is inclined 35° to HP and the base edge at which it is resting is inclined 40° to VP.



B24CS1T01	LOGIC SYSTEM DESIGN	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

The objective of the course is to familiarize learners with the basic concepts of Boolean algebra and digital systems. This course covers the design of simple combinational and sequential logic circuits, representation and arithmetic algorithms for Binary, BCD (Binary Coded Decimal) and floating-point numbers which in turn are helpful in understanding organization & design of a computer system and understanding how patterns of ones and zeros can be used to store information on computers, including multimedia data.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Simplify a given Boolean Function and design a combinational circuit to implement the simplified function using Digital Logic Gates (Cognitive Knowledge level: Apply)
CO 2	Analyze and understand various combinational digital circuits. (Cognitive Knowledge level: Apply)
CO 3	Design of sequential circuits with flip flops.(Cognitive Knowledge level: Apply)
CO 4	Understand the design of shift registers and shifters(Cognitive Knowledge level: Understand)
CO 5	Understand the algorithms of decimal, binary and BCD numbers for performing addition, subtraction, multiplication and division (Cognitive Knowledge level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	1	3	2	1	2		1			
CO 2	2	1	3	2	1	2				1	
CO 3	2	1	3	2	1	2					
CO 4	2	1	3	2	1	2	1		1		2
CO 5	3	1	3	2	1	2		1			

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	35	35	35
Apply	45	45	45
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (8 hours)

Boolean Algebra:

Postulates of Boolean Algebra. Basic theorems and Properties of Boolean Algebra. Boolean Functions - Canonical and Standard forms. Simplification of Boolean Functions- Using Karnaugh- Map Method (up to five variables), Don't care conditions, Product of sums simplification, Tabulation Method.

MODULE 2 (8 hours)

Combinational Logic Circuits :

Design Procedure & Implementation of combinational logic circuits- Binary adders and subtractors, Binary Parallel adder, Carry look ahead adder, Carry save adder, BCD adder, Code converter, Decoder, Demultiplexer, Encoder, Multiplexer, Parity generator/ Checker.

MODULE 3 (9 hours)

Sequential logic circuits:

Flip-flops- SR, JK, T and D. Triggering of flip-flops- Master slave flip- flops, Edge- triggered flip- flops. Excitation table and characteristic equation. Registers- register with parallel load. Counter design: Asynchronous counters- Binary and BCD counters, timing sequences and state diagrams. Synchronous counters- Binary Up- down counter, BCD counter.

MODULE 4 (9 hours)

Shift registers and Verilog:

Shift registers – Serial In Serial Out, Serial In Parallel Out, Bidirectional Shift Register with Parallel load. Ring counter. Johnson counter- timing sequences and state diagrams. Verilog- HDL abstraction; Modern digital design flow- Verilog constructs: data types, modules, Verilog operators. Procedural assignment, Conditional Programming constructs, Test benches, Modelling a D- Flip flop in Verilog.

MODULE 5 (11 hours)

Arithmetic Algorithms:

Algorithms for addition, subtraction, multiplication and division of Decimal number system. Algorithms for addition, subtraction of binary numbers in signed magnitude in 2's complement representations. Algorithms for multiplication, Array multiplier, Booth's multiplication algorithm and division (restoring method) of binary numbers. Algorithms for addition, subtraction, multiplication and division of BCD numbers.

Text Books

1. M. Morris Mano, Digital Logic & Computer Design, 4/e, Pearson Education, 2013 .
2. Thomas L Floyd, Digital Fundamentals, 10/e, Pearson Education, 2009.
3. M. Morris Mano, Computer System Architecture, 3/e, Pearson Education, 2007.
4. Brock J. LaMeres, Introduction to Logic Circuits: Logic Design with Verilog, 2/e, Springer International Publishing, 2017.
5. Michael D. Ciletti, Advanced Digital Design with the Verilog HDL, 2/e, Pearson, 2024.

Reference Books

1. M. Morris Mano, Michael D Ciletti , Digital Design With An Introduction to the Verilog HDL, 5/e, Pearson Education, 2013.
2. Donald D Givone, Digital Principles and Design, Tata McGraw Hill, 2003.
3. Sarah L. Harris, David Harris, Digital Design and Computer Architecture – RISC-V, 1/e, Morgan Kaufmann, 2022.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Boolean Algebra		8 hours
1.1	Introduction to Boolean Algebra: Postulates of Boolean Algebra.	1
1.2	Basic theorems and properties of Boolean Algebra.	1

1.3	Boolean Functions: Canonical and Standard Forms	1
1.4	Simplification of Boolean Functions: Karnaugh Map Method (up to five variables), Don't care conditions	2
1.5	Products of Sum simplification.	1
1.6	Tabulation method	2
Module 2: Combinational Logic Circuits		8 hours
2.1	Design Procedure & Implementation of Combinational Circuits.	1
2.2	Binary Adders: Implementation of Half Adder, Full Adder.	1
2.3	Binary Subtractors: Implementation of Half Subtractor, Full Subtractor.	1
2.4	Implementation of Binary Parallel Adder, Carry look ahead Adder, Carry Save Adder, BCD Adder.	2
2.5	Implementation of Decoder, Demultiplexer.	1
2.6	Implementation of Encoder, Multiplexer.	1
2.7	Implementation of Parity Generator/Checker.	1
Module 3: Sequential logic circuits		9 hours
3.1	Flip flops: SR, JK, T and D flip- flops.	1
3.2	Triggering of flip-flops- Master slave flip- flop, Edge-triggered flip-flops.	1
3.3	Excitation table and characteristic equations of flip- flops.	1
3.4	Registers- Register with parallel load.	1
3.5	Counter Design: Asynchronous counters- Binary and BCD counters- timing sequences and state diagrams.	2
3.6	Asynchronous counters- Binary and BCD counters- timing sequences and state diagrams.	1
3.7	Synchronous counters- Binary Up- down counter, BCD counter.	1
3.8	Using the Delegation Model.	1

Module 4: Shift registers and arithmetic algorithms		9 hours
4.1	Shift Registers - Serial In Serial Out, Serial In Parallel Out	2
4.2	Bidirectional Shift Register with Parallel load.	1
4.3	Shift register counters - Ring Counter.	1
4.4	Johnson Counter- timing sequences and state diagrams.	1
4.5	Verilog: HDL abstraction, modern digital design flow.	1
4.6	Verilog constructs: data types, modules , Verilog operators.	1
4.7	Procedural assignment, conditional programming constructs; test benches.	1
4.8	Modeling a D flip-flop in Verilog.	1
Module 5: Arithmetic Algorithms		11 hours
5.1	Algorithms for addition, subtraction, division and multiplication of decimal number system.	2
5.2	Algorithms for addition, subtraction of binary numbers in signed magnitude in 2's complement representations.	2
5.3	Algorithms for multiplication, array multiplier, Booth's multiplication algorithm	2
5.4	Division (restoring method) of binary numbers.	1
5.5	Algorithm for addition and subtraction of BCD numbers.	2
5.6	Algorithm for multiplication and division of BCD numbers.	2

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

- Given a Boolean function F and don't care conditions D, using Karnaugh map obtain the simplified expression in (i) SOP and (ii) POS:
 - $F(A, B, C, D) = A' B' D' + A'CD + A'BC$
 - $D(A, B, C, D) = A'B C'D + ACD + A B'D$

Course Outcome 2 (CO 2):

1. Design a 4X2 encoder circuit.

Course Outcome 3 (CO 3):

1. Design an asynchronous 3-bit binary down counter.

Course Outcome 4 (CO 4):

1. Design a 4- bit binary Johnson counter.
2. What are Verilog parallel case and full case statements?

Course Outcome 5 (CO 5):

1. Demonstrate multiplication algorithm for BCD numbers.

MODEL QUESTION PAPER

QP CODE:

Pages:3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SECOND SEMESTER B.TECH DEGREE EXAMINATION, JUNE 2025

Course Code: B24CS1T01

Course Name: LOGIC SYSTEM DESIGN

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Find the dual and complement of the boolean function $F = AB' + B(A + B')$.
2. Using K-map, reduce the expression: $A'B + ABC + BC$.
3. Design a half subtractor with NAND gates only.
4. Distinguish between decoder and demultiplexer.
5. Design a combinational circuit that multiplies an input decimal digit by 5 represented in BCD. The output is also in BCD. Show that the outputs can be obtained from the input lines without using any logic gates.
6. Differentiate between synchronous counter and asynchronous counter.
7. Construct D-Flip- flop using NAND gates. Also give its truth table.
8. What is the purpose of procedural assignment in Verilog?
9. Discuss the 4 modes of shift registers.
10. Design a 3x2 array multiplier.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Prove that 8
- i. $AB + A(B + C) + B(B + C) = B + AC$
 - ii. $AB + A(B + C) + B(B + D) = A$
- (b) Using K-map, simplify the Boolean function F in sum of products form, using the don't care conditions d : 6

$$F(w, x, y, z) = w'(x'y + x'y' + xyz) + x'z'(y + w)$$

$$d(w, x, y, z) = w'x(y'z + y'z') + wy$$

OR

12. (a) Simplify the following expressions using Karnaugh-map method. 6
- i. $F = \Sigma(0, 2, 4, 6, 9, 11, 13, 15, 17, 21, 25, 27, 29, 31)$
 - ii. $F = \Pi(0, 2, 5, 7)$
- (b) Convert the following to the other canonical form: 8
- i. $F(x, y, z, a) = (1, 3, 7)$
 - ii. $F(x, y, z) = \Pi(0, 3, 6, 7)$
 - iii. $F(A, B, C, D) = \Pi(0, 1, 2, 3, 4, 6, 12)$
13. (a) Implement Full adder circuit using NAND gate only. 6
- (b) Discuss the design and timing sequence of a BCD counter with a suitable example. 8

OR

14. (a) With a neat diagram explain 4-bit carry look-ahead adder. 6
- (b) Explain the construction and working of a 4-bit binary adder-subtractor using full adders and XOR gates (for 2's complement subtraction). 8
15. (a) Explain in detail the working and applications of SR, JK, T, and D flip-flops with neat diagrams 8
- (b) Explain the design procedure for synchronous counters. Design a 4-bit synchronous binary up counter using JK flip-flops. Draw the circuit diagram, excitation table derivation, K-maps (if needed), and state diagram. 6

OR

16. (a) Write the characteristic equation and excitation table of JK, T and D flip-flops. 6
- (b) Design a synchronous Binary Up-Down Counter. 8
17. (a) Describe a Johnson counter (twisted ring counter). How many states does a 4-bit Johnson counter have, and what is its modulus? 6

- (b) Describe Verilog constructs in detail: modules, ports, data types (wire, reg, integer), and operators (arithmetic, relational, logical, bitwise, reduction). Provide suitable code examples for each. 8

OR

18. (a) What are the data types in Verilog? Explain the uses of Verilog. 6
(b) Explain the modern digital design flow using Verilog HDL. Discuss the role of Verilog in each stage (specification, RTL design, synthesis, simulation, verification) with a block diagram of the flow. 8
19. (a) Perform -9×13 multiplication using Booth algorithm. 7
(b) Write the algorithm for multiplication of BCD numbers. 7

OR

20. (a) Explain the restoring division with proper example. 7
(b) Design a 4x3 array multiplier. 7

B24CS1T02	INDUSTRIAL PROGRAMMING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

The objective of the course is to equip the learners to develop multi-module software solutions for real-world computational problems using Python. It encompasses the Python programming environment, syntax, data representations, intermediate level features, GUI programming, Object Oriented Programming and data processing. This course lays the foundation to develop modular software solutions including complex interactive applications, network applications, and data-driven intelligent applications.

Prerequisites

Basic knowledge in Computational Problem Solving, A course in any programming language

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Write, test, and debug Python programs, applying engineering fundamentals and natural science principles (e.g., data representation, algorithmic thinking) to solve computational problems. (Cognitive Knowledge level: Apply)
CO 2	Develop programs utilizing Python programming constructs such as Lists, Tuples, Sets, and Dictionaries, using mathematical and statistical concepts for data organization and manipulation. (Cognitive Knowledge level: Apply)
CO 3	Develop graphical user interfaces for solutions using Python libraries (Turtle, GUI frameworks), while considering resource efficiency and user-centered design. (Cognitive Knowledge level: Apply)
CO 4	Implement Object-Oriented programs with exception handling, demonstrating ethical and inclusive practices in software development (e.g., accessible design, error handling). (Cognitive Knowledge level: Apply)
CO 5	Write programs in Python to process data stored in files by utilizing NumPy, Matplotlib, and Pandas, while engaging with current research literature and evaluating societal implications of data-driven solutions (e.g., privacy, bias). (Cognitive Knowledge level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	1	2	3						1
CO 2	2	3	1	2	3						
CO 3	2	3	2	2	3	1				2	
CO 4	2	3	2	2	3	1	3	3			
CO 5	2	3	2	3	3	3			3		3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	30	30	30
Apply	50	50	50
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (5 hours)

Programming Environment and Python Basics:

Getting started with Python programming – Interactive shell, IDLE, IPython Notebooks, Detecting and correcting syntax errors, How Python works. The software development process –A case study. Basic coding skills – strings, assignment, and comments, Numeric data types and character sets, Expressions, Using inbuilt functions and modules. Control statements – Iteration with for/while loop, Formatting text for output- A case study. Selection structure (if-else, switch case), Conditional iteration with while - A case study, Testing control statements, Lazy evaluation.

MODULE 2 (8 hours)

Building Python Programs:

Strings and text files – Accessing characters, substrings, Data encryption, Strings and number system, String methods, Text files, A case study on text analysis. Design with Functions – Functions as Abstraction Mechanisms, Problem solving with top-down design, Design with recursive functions, Managing a program's namespace, Higher-Order Functions. Lists - Basic list Operations and functions, List of lists, Slicing, Searching and sorting list, List comprehension. Work with tuples. Sets. Work with dates and times, A case study with lists. Dictionaries - Dictionary functions, dictionary literals, adding and removing keys, accessing and replacing values, traversing dictionaries, reverse lookup. Case Study – Data Structure Selection.

MODULE 3 (6 hours)

Graphics:

Terminal-based programs, Simple Graphics using Turtle, Operations, 2D Shapes, Colors and RGB Systems, A case study. Image Processing – Basic image processing with inbuilt functions. Graphical User Interfaces – Event-driven programming, Coding simple GUI-based programs : Windows, Labels, Displaying images, Input text entry, Popup dialog boxes, Command buttons, A case study.

MODULE 4 (7 hours)

Object Oriented Programming:

Design with classes - Objects and Classes, Methods, Instance variables, Constructor, Accessor and Mutator, Data-Modeling Examples, Structuring classes with inheritance and polymorphism. Abstract classes, Interfaces, Exceptions Handle a single exception, handle multiple exceptions.

MODULE 5 (9 hours)

Data Processing:

Python built-in modules - “os”, “sys”, NumPy - Basics, Creating arrays, Arithmetic, Slicing, Matrix Operations, Random numbers. Plotting and visualization-Matplotlib - Basic plot, Ticks, Labels, and Legends. Working with CSV files – Pandas Reading, Manipulating, and Processing Data. Introduction to Micro services using Flask.

Text Books

1. Kenneth A Lambert., Fundamentals of Python: First Programs, 2/e, Cengage Publishing, 2016
2. Wes McKinney, Python for Data Analysis, 2/e, Shroff / O'Reilly Publishers, 2017
3. Flask: Building Python web services, Jack Stouffer, Shalabh Aggarwal, Gareth Dwyer, PACKT Publishing Limited, 2018

Reference Books

4. Zed A Shaw, Learn Python 3 The Hard Way, Addison-Wesley, 2017
5. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, 2/e, Shroff, 2016
6. Michael Urban and Joel Murach, Python Programming, Shroff/Murach, 2016
7. Charles Severance, Python for Informatics: Exploring Information,

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		35 Hours
Module 1: Programming Environment and Python Basics		5 hours
1.1	Getting started with Python programming – Interactive shell, IDLE, iPython Notebooks, Detecting and correcting syntax errors, How Python works.	1
1.2	The software development process – A case study.	1
1.3	Basic coding skills – strings, assignment, and comments, Numeric data types and character sets, Expressions, Using inbuilt functions and modules.	1
1.4	Control statements – Definite Iteration with for loop, Formatting text for output, Selection structure (if-else, switch-case), Conditional iteration with while loop, A case study	1
1.5	Testing the control statements, Lazy evaluation.	1
Module 2: Building Python Programs		8 hours
2.1	Strings – Accessing characters, substrings, Data encryption, Strings and number system, String methods,	1
2.2	Text files, A case study on text analysis.	1
2.3	Design with Functions – Functions as Abstraction Mechanisms, Problem solving with top-down design,	1
2.4	Design with recursive functions, Managing a program's namespace, Higher- Order Functions.	1
2.5	Lists - Basic list Operations and functions, List of lists, Slicing, Searching and sorting list, List comprehension.	1
2.6	Work with tuples. Sets. Work with dates and times, A case study with lists.	1
2.7	Dictionaries - Dictionary functions, dictionary literals, adding and removing keys, accessing and replacing values, traversing dictionaries, reverse lookup.	1
2.8	Case Study - Data Structure Selection.	1
Module 3: Graphics		6 hours
3.1	Graphics – Simple Graphics using Turtle, Operations, 2D Shapes, Colors and RGB Systems, A case study.	1

3.2	Image Processing – Basic image processing with inbuilt functions.	1
3.3	Graphical User Interfaces – Event-driven programming	1
3.4	Coding simple GUI-based programs: Windows, Labels, Displaying images,	1
3.5	Coding simple GUI-based programs: Input text entry, Popup dialog boxes, Command buttons	1
3.6	A case study - GUI	1
Module 4: Object Oriented Programming		7 hours
4.1	Design with classes: Objects and Classes, Methods, Instance Variables	1
4.2	Constructor, Accessors, and Mutators	1
4.3	Structuring classes with Inheritance	1
4.4	Polymorphism	1
4.5	Abstract classes	1
4.6	Interfaces	1
4.7	Exceptions: Handle a single exception, handle multiple exceptions	1
Module 5: Data Processing		9 hours
5.1	The os and sys modules, NumPy : Basics, Creating arrays, Arithmetic, Slicing	1
5.2	Matrix Operations, Random numbers.	1
5.3	Matplotlib : Basic plot, Ticks, Labels, and Legends	1
5.4	Working with CSV files	1
5.5	Pandas : Reading, Manipulating	1
5.6	Pandas : Processing Data and Visualize.	1
5.7	Introduction to Microservices using Flask	1
5.8	Introduction to Microservices using Flask	1
5.9	Introduction to Microservices using Flask	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. What is type conversion? How is it done in Python?
2. Write a note on the Python editors.
3. What is printed when the below code is executed?

```
mysum = 0
for i in range(5, 11, 2):
    mysum += i
    if mysum == 5:
        break
    mysum += 1
print(mysum)
```

What would be the output if 'break' is replaced with 'continue' in the above code fragment?

Course Outcome 2 (CO 2):

1. Given is a list of words, wordlist, and a string, name. Write a Python function which takes wordlist and name as input and returns a tuple. The first element of the output tuple is the number of words in the wordlist which have name as a substring in it. The second element of the tuple is a list showing the index at which the name occurs in each of the words of the wordlist and a 0 if it doesn't occur
2. What is the value of L after you run the code below?

```
L = ["life", "answer", 42, 0]
for thing in L:
    if thing == 0:
        L[thing] = "universe"
    elif thing == 42:
        L[1] = "everything"
```

3. Illustrate how to change a key from a dictionary

Course Outcome 3 (CO 3):

1. A bouncy program is defined as follows – The program computes and displays the total distance travelled by a ball, given three inputs—the initial height from which it is dropped, its bounciness index, and the number of bounces. Given the inputs write a GUI-based program to compute the total distance traveled.
2. Write a Python program to find the quadrant of a point, say (x,y).

3. Write a program to change a color image to black and white.

Course Outcome 4 (CO 4):

1. Write a Python program to implement the addition, subtraction, and multiplication of complex numbers using classes. Use constructors to create objects. The input to the program consist of real and imaginary parts of the complex numbers.
2. Explain inheritance in Python using suitable examples.
3. Illustrate the use of exceptions.

Course Outcome 5 (CO 5):

1. Given a file “auto.csv” of automobile data with the fields index, company, body-style, wheelbase, length, engine-type, num-of-cylinders, horsepower, average-mileage, and price, Write a python code to
 - (i) Clean and Update the CSV file
 - (ii) Print total cars of all companies
 - (iii) Find the average mileage of all companies
 - (iv) Find the highest priced car of all companies.
2. Given two matrices A and B, write a program to find the product of A and B.
3. Write a program to plot the histogram of an image.

MODEL QUESTION PAPER

QP CODE:

Pages: 4

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SECOND SEMESTER B.TECH DEGREE EXAMINATION, JUNE 2025

Course Code: B24CS1T02

Course Name: INDUSTRIAL PROGRAMMING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Write a Python program to reverse a number and also find the sum of digits of the number. Prompt the user for input.
2. Explain the concept of scope and lifetime of variables in Python programming language, with a suitable example.
3. Illustrate format specifiers and escape sequences with examples.
4. Compare tuples, lists, and dictionaries with examples. Describe the following dictionary methods with an example.
5. Describe the following dictionary methods with example:
 - i. `get( )`
 - ii. `Keys( )`
 - iii. `pop( )`
 - iv. `update( )`
 - v. `values( )`
 - vi. `items( )`
6. Differentiate the terminal-based and GUI-based programming in Python.

7. What is polymorphism? Give an example in the context of OOP in Python.
8. How is exception handling accomplished in Python programs?
9. Explain the os and os.path modules in Python with examples. Also discuss walk() and getcwd() methods of the os module.
10. What are the important characteristics of CSV file format?

PART B

Answer any one question from each module. Each question carries 14 marks.

1. (a) Write a Python code to check whether a given year is a leap year or not [A year is a leap year if it's divisible by 4 but not divisible by 100 except for those divisible by 400]. 6
(b) What are the possible errors in a Python program. Write a Python program to print the value of $22n + n + 5$ for n provided by the user. 8

OR

2. (a) Write a Python program to find the value for $\sin(x)$ up to n terms using the series
$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots$$
where x is in degrees. 6
(b) Write a Python code to determine whether the given string is a Palindrome or not using slicing. Do not use any string function. 8
3. (a) Write a Python code to create a function called `list_of_frequency` that takes a string and prints the letters in non-increasing order of the frequency of their occurrences. Use dictionaries. 5
(b) Write a Python program to read a list of numbers & sort the list in ascending order without using any built-in functions. A separate function should be written to sort the list where the name of the list is passed as the parameter. 9

OR

4. (a) Illustrate the following Set methods with an example. 6
 - i. `intersection()`
 - ii. `union()`
 - iii. `issubset()`
 - iv. `difference()`
 - v. `update()`
 - vi. `discard()`
- (b) Write a Python program to check the validity of a password given by the user. The Password should satisfy the following criteria: 8

- i. Contains at least one letter between a and z
 - ii. Contains at least one number between 0 and 9
 - iii. Contains at least one letter between A and Z
 - iv. Contains at least one special character from \$, #,
 - v. Minimum length of password: 6
5. (a) Write a program to draw a hexagon using turtle. 5
- (b) Write a note on the image processing function in Python. 9

OR

6. (a) Describe the features of event-driven programming. 4
- (b) Write a GUI-based program that allows the user to convert temperature values between degrees Fahrenheit and degrees Celsius. The interface should have labeled entry fields for these two values. These components should be arranged in a grid where the labels occupy the first row and the corresponding fields occupy the second row. At start-up, the Fahrenheit field should contain 32.0, and the Celsius field should contain 0.0. The third row in the window contains two command buttons, labeled »» and ««. When the user presses the first button, the program should use the data in the Fahrenheit field to compute the Celsius value, which should then be output to the Celsius field. The second button should perform the inverse function. 10
7. (a) How can a class be instantiated in Python? Write a Python program to express the instances as return values to define a class RECTANGLE with parameters height, width, corner_x, and corner_y and member functions to find center, area, and perimeter of an instance. 10
- (b) Explain inheritance in Python. Give examples for each type of inheritance. 4

OR

8. (a) Write a Python class named Circle constructed by a radius and two methods which will compute the area and the perimeter of a given circle. 6
- (b) Write Python program to create a class 'Complex' and implement __add__() method to add two complex numbers. Display the result by overloading the + operator. 8
9. (a) Write a Python program to add two matrices and also find the transpose of the resultant matrix. 8
- (b) Given a file "auto.csv" of automobile data with the fields index, company, body-style, wheel-base, length, engine-type, num-of-cylinders, horsepower, average-mileage, and price, write Python codes using Pandas to 6
- i. Clean and Update the CSV file
 - ii. Print total cars of all companies
 - iii. Find the average mileage of all companies
 - iv. Find the highest priced car of all companies.

OR

10. (a) Write Python program to write the data given below to a CSV file.

5

SN	Name	Country	Contribution	Year
1	Linus Tovalds	Finland	Linux Kernal	1991
2	Tim Berners Lee	England	World Wide Web	1990
3	Guido Van Rossum	Netherlands	3 Python	1991

(b) Given the sales information of a company as CSV file with the following fields month_number, facecream, facewash, toothpaste, bathingsoap, shampoo, moisturizer, total_units, total_profit. Write Python codes to visualize the data as follows: 9

- Toothpaste sales data of each month and show it using a scatter plot.
- Face cream and face wash product sales data and show it using the bar chart. Calculate total sale data for last year for each product and show it using a Pie chart.

B24CS1T03	OBJECT ORIENTED PROGRAMMING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

The purpose of this course is to enable learners to solve problems by breaking it down to object level while designing software and to implement it using Java. This course covers Object Oriented Principles, Object Oriented Programming in Java, Inheritance, Exception handling, Event handling, multithreaded programming, basics of networking with Java and working with window-based graphics. This course helps the learners to develop Desktop GUI Applications, Mobile applications, Enterprise Applications, Scientific Applications and Web based Applications.

Prerequisites

Basic knowledge in Computational Problem Solving, A course in any programming language

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Identify the fundamentals of client/server systems, server types, characteristics, and the driving forces behind their development. (Cognitive Knowledge Level: Understand)
CO 2	Explain client and server components, hardware platforms, operating systems, and communication using the OSI model. (Cognitive Knowledge Level: Understand)
CO 3	Apply middleware concepts, RPC mechanisms, and ODBC architecture to select appropriate connectivity solutions for distributed client/server applications. (Cognitive Knowledge Level: Apply)
CO 4	Apply system management techniques such as remote administration tools, performance monitoring, and scalability strategies to maintain efficient client/server systems. (Cognitive Knowledge Level: Apply)
CO 5	Apply client/server database architectures and middleware components to design suitable database-driven client/server solutions. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	2		1		2	2	2	1	1
CO 2	2	3	3		2	1	2	2	3	1	2
CO 3	2	3	3	2	2	3	3	2	2	2	2
CO 4	2	3	3	3	3	3	3	3	3	3	3
CO 5	1	3	3	2	2	3	1	3	3	2	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignments)	5 marks
Assignment/Analytical	

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (8 hours)

Introduction:

Approaches to Software Design - Functional Oriented Design, Object Oriented Design. Basic Object-Oriented Concepts, Object Modeling Using Unified Modeling Language (UML)- UML diagrams- Use case model, Class diagram, Interaction diagram.

Introduction to Java- Java programming Environment and Runtime Environment, Development Platforms -Standard, Enterprise. Java Virtual Machine (JVM), Java compiler, Bytecode, Java applet, Java Buzzwords, Java program structure, Garbage Collection, Lexical Issues.

Primitive Data types - Integers, Floating Point Types, Characters, Boolean. Literals, Type Conversion and Casting, Variables, Arrays, Strings, Vector class.

Operators - Arithmetic Operators, Bitwise Operators, Relational Operators, Boolean Logical Operators, Assignment Operator, Conditional (Ternary) Operator, Operator Precedence.

Control Statements - Selection Statements, Iteration Statements and Jump Statements.

MODULE 2 (10 hours)

Object Oriented Programming in Java:

Class Fundamentals, Declaring Objects, Object Reference, Introduction to Methods, Constructors, this Keyword, Method Overloading, Using Objects as Parameters, Returning Objects, Recursion, Access Control, Static Members, Final Variables, Inner Classes, Command Line Arguments, Variable Length Arguments.

Inheritance - Super Class, Sub Class, The Keyword super, protected Members, Calling Order of Constructors, Method Overriding, the Object class, Abstract Classes and Methods, using final with Inheritance

String handling: Special String Operations - Character Extraction, String Comparison, Searching Strings, Modifying Strings, using valueOf(), Comparison of StringBuffer and String.

MODULE 3 (12 hours)

More Features of Java:

Packages and Interfaces - Defining Package, CLASSPATH, Access Protection, Importing Packages, Interfaces.

Exception Handling - Checked Exceptions, Unchecked Exceptions, try Block and catch Clause, Multiple catch Clauses, Nested try Statements, throw, throws and finally.

Input/Output - I/O Basics, Reading Console Input, Writing Console Output, PrintWriter Class, Object Streams and Serialization, Working with Files.

Network Programming with Java – Socket Programming in Java, Client Sockets, Server Sockets, Data Transmission through sockets, TCP/IP Programming with Java, Datagram Packet, Datagram Server and Client, A client/Server Example, Sending and receiving objects.

MODULE 4 (8 hours)

Advanced features of Java:

Collections framework - Collections overview, Collections Interfaces- Collection Interface, List Interface. Collections Class – ArrayList class. Accessing a Collection via an Iterator. Multithreaded Programming - The Java Thread Model, The Main Thread, Creating Thread, Creating Multiple Threads, Synchronization, Suspending, Resuming and Stopping Threads. Event handling - Event Handling Mechanisms, Delegation Event Model, Event Classes, Sources of Events, Event Listener Interfaces, Using the Delegation Model.

MODULE 5 (7 hours)

Graphical User Interface and Database Support of Java:

Swing fundamentals - Swing Key Features, Model View Controller (MVC), Swing Controls, Components and Containers, Swing Packages, Event Handling in Swing, Swing Layout Managers, Exploring Swing –JFrame, JLabel, The Swing Buttons, JTextField.

Java DataBase Connectivity (JDBC) - JDBC overview, Creating and Executing Queries – create table, delete, insert, select.

Text Books

1. Herbert Schildt, Java: The Complete Reference, 8/e, Tata McGraw Hill, 2011.
2. Paul Deitel, Harvey Deitel, Java How to Program, Early Objects 11th Edition, Pearson, 2018.
3. Y. Daniel Liang, Introduction to Java Programming, 7/e, Pearson, 2013.

Reference Books

1. Nageswararao R., Core Java: An Integrated Approach, Dreamtech Press, 2008
2. Flanagan D., Java in A Nutshell, 5/e, O'Reilly, 2005.
3. Barclay K., J. Savage, Object Oriented Design with UML and Java, Elsevier, 2004.
4. Sierra K., Head First Java, 2/e, O'Reilly, 2005.
5. Balagurusamy E., Programming JAVA a Primer, 5/e, McGraw Hill, 2014.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction		8 hours
1.1	Approaches to Software Design - Functional Oriented Design, Object Oriented Design. Basic Object-Oriented Concepts.	1
1.2	Object Modeling Using Unified Modeling Language (UML)- UML diagrams- Use case model, Class diagram, Interaction diagram.	1
1.3	Java programming Environment and Runtime Environment, Development Platforms -Standard, Enterprise. JVM, Java compiler, Bytecode.	1
1.4	Java applet, Java Buzzwords, Java program structure, Comments, Garbage Collection, Lexical Issues.	1
1.5	Primitive Data types - Integers, Floating Point Types, Characters, Boolean.	1
1.6	JLiterals, Type Conversion and Casting, Variables, Arrays, Strings, Vector class.	1
1.7	Operators - Arithmetic Operators, Bitwise Operators, Relational Operators, Boolean Logical Operators, Assignment Operator, Conditional (Ternary) Operator, Operator Precedence.	1
1.8	Control Statements - Selection Statements, Iteration Statements and Jump Statements.	1
Module 2: Object Oriented Programming in Java		10 hours
2.1	Class Fundamentals, Declaring Objects, Object Reference.	1
2.2	Introduction to Methods, Constructors, this Keyword, Method Overloading, Using Objects as Parameters.	1
2.3	Returning Objects, Recursion, Access Control.	1
2.4	Static Members, Final Variables, Inner Classes, Command Line Arguments, Variable Length Arguments.	1
2.5	Inheritance - Super Class, Sub Class, The Keyword super, protected Members.	1

2.6	Calling Order of Constructors, Method Overriding, the Object class.	1
2.7	Calling Order of Constructors, Method Overriding, the Object class.	1
2.8	String Handling – String Constructors, String Length.	1
2.9	String handling: Special String Operations - Character Extraction, String Comparison, Searching Strings.	1
2.10	Modifying Strings, using valueOf(), Comparison of StringBuffer and String.	1
Module 3: More Features of Java		12 hours
3.1	Packages and Interfaces - Defining Package, CLASSPATH, Access Protection, Importing Packages.	1
3.2	Interfaces.	1
3.3	Exception Handling - Checked Exceptions, Unchecked Exceptions, try Block and catch Clause.	
3.4	Multiple catch Clauses, Nested try Statements. Throw, throws and finally.	1
3.5	Input/Output - I/O Basics, Reading Console Input, Writing Console Output, PrintWriter Class.	1
3.6	Object Streams and Serialization.	1
3.7	Working with Files.	1
3.8	Network Programming with Java – Socket Programming in Java, Client Sockets, Server Sockets.	1
3.9	Data Transmission through sockets, TCP/IP Programming with Java.	1
3.10	Datagram Packet, Datagram Server and Client.	1
3.11	A client/Server Example.	1
3.12	Sending and receiving objects.	1
Module 4: Advanced features of Java		8 hours
4.1	Collections framework - Collections overview, Collections Interfaces- Collection Interface.	1
4.2	List Interface, Collections Class – ArrayList class.	1
4.3	Accessing a Collection via an Iterator.	1
4.4	Advanced features of Java: Multithreaded Programming - The Java Thread Model, The Main Thread, Creating Thread.	1

4.5	Creating Multiple Threads, Synchronization, Suspending, Resuming and Stopping Threads.	1
4.6	Event handling - Event Handling Mechanisms, Delegation Event Model.	1
4.7	Event Classes, Sources of Events, Event Listener Interfaces.	1
4.8	Using the Delegation Model.	1
Module 5: Graphical User Interface and Database Support of Java		7 hours
5.1	Swing fundamentals - Swing Key Features, Model View Controller (MVC).	1
5.2	Swing Controls, Components and Containers.	1
5.3	Swing Packages, Event Handling in Swing.	1
5.4	Swing Layout Managers.	1
5.5	Exploring Swing –JFrame, JLabel, The Swing Buttons, JTextField.	1
5.6	Java DataBase Connectivity (JDBC) - JDBC overview.	1
5.7	Creating and Executing Queries – create table, delete, insert, select.(Basics only, DBMS Course is not a prerequisite)	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. How is platform independence achieved in Java?
2. Discuss the advantages and disadvantages of OOD.
3. Compare and contrast Java Applets and Java Application.

Course Outcome 2 (CO 2):

1. Write a Java program to evaluate a post fix expression containing two operands and a single operator using stack. Stack should be implemented as a separate entity so as to reflect OOP concepts.
2. Demonstrate with example, one method each, from the String class, which performs the below special string operations: Character Extraction, Searching Strings and Modifying Strings.

3. Summarize the need of constructors in Java and list the different types of constructors allowed in Java programs.

Course Outcome 3 (CO 3):

1. Write a program in java to read a paragraph from a text file and print to console.
2. Demonstrate the significance of the keywords 'try', 'catch', 'finally', 'throw' and 'throws' in exception handling of Java with appropriate examples.
3. How do you create and import a package in Java?
4. Explain accept() function in socket programming.

Course Outcome 4 (CO 4):

1. Write a program to demonstrate the start, run, sleep and join methods in Thread class.
2. Write a Java program that creates multiple child threads to print odd and even numbers from 50-100
3. Write a Java program to compare two linked lists. List 1 has the names of 5 colours and list 2 has 4 colours. Print the colours that are not present in both.

Course Outcome 5 (CO 5):

1. Write a GUI based program with separate buttons to add, delete and display student details i.e. name, student ID, current semester and branch of study based on student ID.
2. Using Swing create a JFrame with a JLabel and JButtons. Set the texts of JButtons as "Yes" and "No" respectively. Set the JLabel's text to the text of the Button currently being pressed. Initially the JLabel's text is blank.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SECOND SEMESTER B.TECH DEGREE EXAMINATION, JUNE 2025

Course Code: B24CS1T03

Course Name: OBJECT ORIENTED PROGRAMMING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Briefly explain the portable, secure and robust features of Java.
2. Describe the concepts of object and class with a suitable Java program.
3. Explain the concept of method overriding with an example.
4. What is the use of the keyword final in Java?
5. Explain the concept of streams.
6. Explain any two applications of Serialization.
7. List any six event listener interfaces.
8. What are Collections in Java? Explain any one Collection interface in Java.
9. Explain any two properties of Swing components in Java.
10. Explain JLabel component. With suitable examples explain any two of its constructors.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Describe in detail any three Object Oriented Programming principles. Illustrate with suitable examples. 9
(b) What is Java Runtime Environment? What is the role of Java Virtual Machine in it? 5

OR

12. (a) Explain automatic type conversion in Java with an example. What are the two conditions required for it? 6
(b) Explain in detail the primitive data types in Java. 8
13. (a) Using a suitable Java program explain the difference between private and public members in the context of inheritance. 8
(b) Is it possible to use the keyword super within a static method? Give justification for your answer. 6

OR

14. (a) Outline the significance of String class in Java and list any of its five built in functions. 5
(b) Outline the significance of String class in Java and list any of its five built in functions' 9
15. (a) Explain in detail about byte streams and character streams with suitable code samples. 6
(b) Describe in detail about exception handling, try block and catch clause with the help of a suitable Java program. 8

OR

16. (a) Write a program to establish socket connection between a client and server. Let the client enter two numbers to the server. The server adds the numbers and send back the sum to the client. 8
(b) Explain throw, throws and finally constructs with the help of a Java program. 6
17. (a) HDescribe in detail the creation of a thread using the Runnable interface and the Thread class with suitable examples. 10
(b) Explain List Interface. Mention any two exceptions thrown by its methods. 4

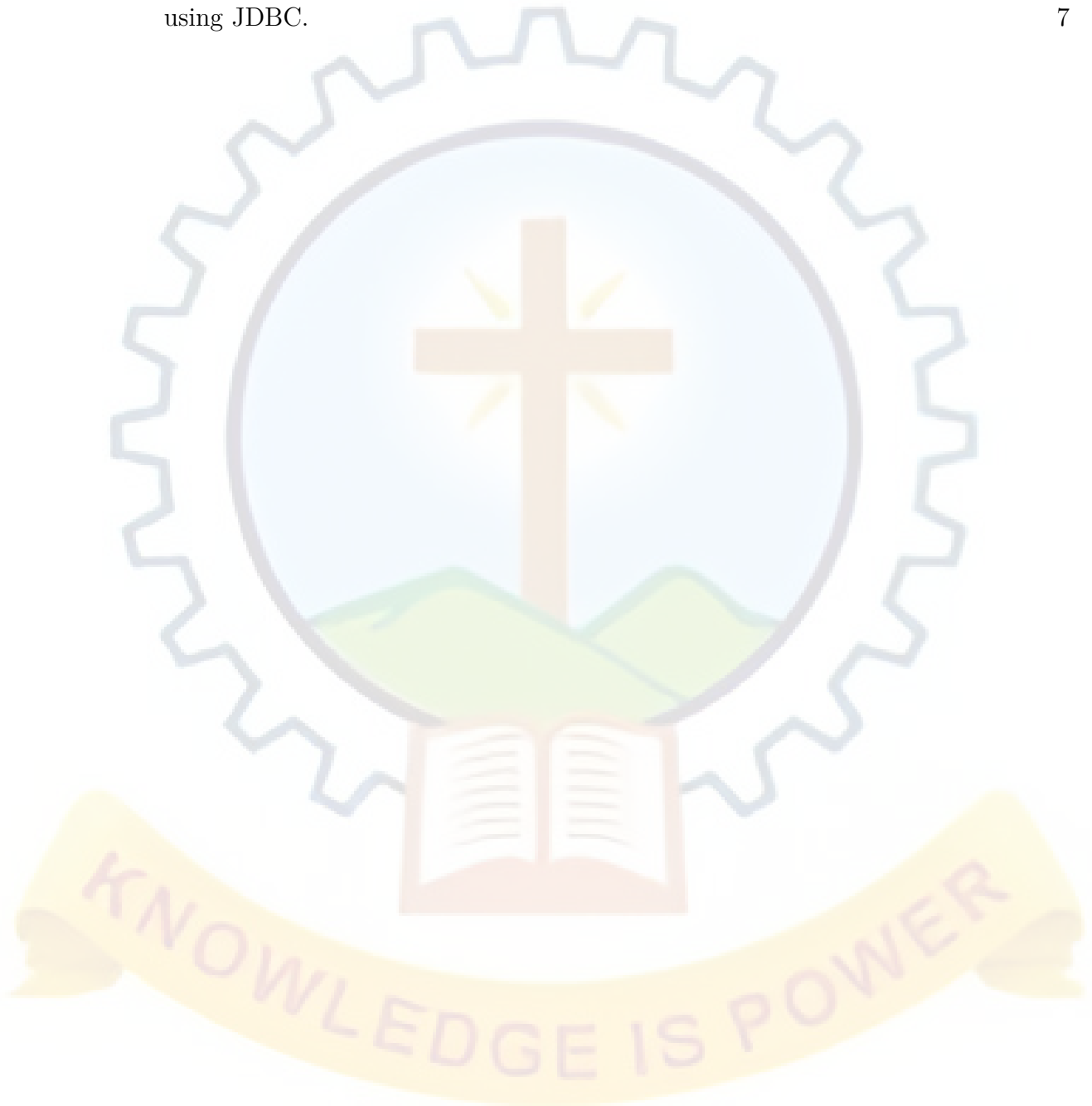
OR

18. (a) Explain in detail the Delegation Event model for event handling in Java. 7
(b) Write a simple program by extending appropriate class to demonstrate the working of threads in java. 7
19. (a) Write a Java program to demonstrate the use of JLabel and JButton by adding them to JFrame. 7

- (b) Explain the step-by-step procedure of using Java Database Connectivity in Java programs. 7

OR

20. (a) Explain the class hierarchy of Java Swing components. 7
(b) Write a Java Program to create a student table and to add student details to it using JDBC. 7



B24CS1L01	INDUSTRIAL PROGRAMMING LAB	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

The aim of the course is to equip the learners to develop multi-module software solutions for real world computational problems using Python. The students will have a hands-on experience in Python programming environment, syntax, data representations, intermediate level features, GUI programming, Object Oriented Programming and data processing. After the lab session student will be able to design, implement, test, and deploy robust and scalable Python applications for various industrial applications.

Prerequisite

Basic knowledge in Computational Problem Solving

Course Outcomes

After the completion of the course the student will be able to

CO 1	Experiment with concepts of iteration, function, string, and list, applying engineering fundamentals and natural science principles (e.g., data representation, algorithmic thinking) to solve computational problems through hands-on programming.(Cognitive Knowledge level: Apply)
CO 2	Develop Python programs which use tuples, dictionary traversal, dictionary methods, files, and operations, using mathematical and statistical concepts for data organization and manipulation in real-world datasets.(Cognitive Knowledge level: Apply)
CO 3	Design turtle graphics and develop graphical user interfaces for solutions, while considering resource efficiency, user-centered design, and reproducibility in software development.(Cognitive Knowledge level: Apply)
CO 4	Implement Object-Oriented programs with exception handling, demonstrating ethical and inclusive practices in software development (e.g., accessible design, robust error handling, collaborative teamwork).(Cognitive Knowledge level: Apply)
CO 5	Develop programs in Python to process data stored in files by utilizing NumPy, Matplotlib, and Pandas, while engaging with current research literature and evaluating societal implications of data-driven solutions (e.g., privacy, bias, responsible data usage).(Cognitive Knowledge level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	1	2	3						1
CO 2	2	3	1	2	3						
CO 3	2	3	2	2	3	1				2	
CO 4	1	3	2	2	3	1	3	3			
CO 5	2	3	2	3	3	3			3		3

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Evaluation	20 marks
- Viva (Self Study)	10 marks
- Output	10 marks
Lab Record	5 marks
Viva-voce/Test	15 marks

SYLLABUS

LIST OF EXPERIMENTS

1	Write a program to find the largest of three numbers.
2	Write a program to print the multiplication table of a number n.
3	Write a program to replace a word by another word in a sentence.
4	Write a program to find Surface area and volume of a cylinder using function.

5	Write a program to confirm the validity of an email id by verifying its format.
6	Write a program to remove duplicates from a list.
7	Write a program to add two matrices.
8	Write a program to read a tuple of numbers and print even tuple and odd tuple.
9	Create a dictionary with a set of book title and corresponding stock. Write a program to update the stock and to add or delete books.
10	A set of numbers are stored in a file. Write a program to print the prime numbers among them.
11	Write a program to count the number of words, sentences, upper case letters, lowercase letters and special symbols in a text stored in file.
12	The Payroll Department keeps a list of employee information for each pay period in a text file. The format of each line of the file is the following: <last name> <hourly wage> <hours worked> Write a program that inputs a filename from the user and prints to the terminal a report of the wages paid to the employees for the given period. The report should be in tabular format with the appropriate header. Each line should contain an employee's name, the hours worked, and the wages paid for that period.
13	Write a Python program to find the value for $\sin(x)$ up to n terms using the series using functions $\sin(x) = \frac{x}{1!} - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots$ where x is in degrees.
14	Write a recursive function that expects a pathname as an argument. The pathname can be either the name of a file or the name of a directory. If the pathname refers to a file, its name is displayed, followed by its contents. Otherwise, if the pathname refers to a directory, the function is applied to each name in the directory.
15	Higher order functions: - Write the code for a mapping that generates a list of the absolute values of the numbers in a list named numbers. - Write the code for a filtering that generates a list of the positive numbers in a list named numbers. You should use a lambda to create the auxiliary function. - Write the code for a reducing that creates a single string from a list of strings named words.

16	Turtle graphics: a) Write a menu driven program to draw turtle graphics circle, triangle, square, rectangle, pentagon and hexagon. b) Write a python function to read a shape and display a radial pattern of that shape
17	GUI Programs: a) Temperature Converter: Create a GUI application that allows users to convert temperature between Celsius and Fahrenheit. b) Simple Calculator: Develop a basic calculator application with buttons for arithmetic operations (+, -, *, /) and display for input and output. c) Image Viewer: Create a simple image viewer application that allows users to browse and display images from their computer.
18	Object Oriented Programs: a) Define a student class with attributes such as name, age, and grade. Include methods to calculate the students average grade and display student details. b) Develop a menu-driven program to perform addition, subtraction, and other arithmetic operations on complex numbers. c) Create a menu-driven application for executing arithmetic operations on rational numbers. Employ an exception handling mechanism to manage potential errors. d) Design and implement a blackjack game using Python. The game should allow one player to play against the computerized dealer. The player should be able to hit, stand, double down, and split according to the standard rules of blackjack. Additionally, implement betting functionality where players can place bets before each hand. The game should display the player's hand, the dealer's visible card, and the current balance. Ensure that the game adheres to standard blackjack rules and provides an interactive and enjoyable gaming experience for the player.
19	Data Visualization Using Matplotlib: a) Develop a program to plot a simple line graph using Matplotlib, with data read from a CSV file using Pandas. b) Write a Python script to create a scatter plot of two variables from a CSV file and customize the plot by adding labels, title, and legend using Matplotlib.

20	NumPy: a) Implement a program to generate random data arrays using NumPy and save them to a text file. Then, read the data back from the file and visualize it using Matplotlib. b) Write a Python program to read data from a CSV file using NumPy and perform basic statistical analysis (mean, median, standard deviation, etc.) on the data. c) Implement Matrix multiplication
21	Pandas: a) Create a Python program to read data from an Excel file using Pandas and perform data manipulation operations like filtering, sorting, and grouping. b) Implement a program to merge data from multiple CSV files using Pandas and save the merged data to a new CSV file. Then, visualize the merged data using Matplotlib.

Reference Books

1. Kenneth A Lambert., Fundamentals of Python: First Programs, 2/e, Cengage Publishing, 2016
2. Wes McKinney, Python for Data Analysis, 2/e, Shroff / O'Reilly Publishers, 2017
3. Flask: Building Python web services, Jack Stouffer, Shalabh Aggarwal, Gareth Dwyer, PACKT Publishing Limited, 2018
4. Zed A Shaw, Learn Python 3 The Hard Way, Addison-Wesley, 2017
5. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, 2/e, Schroff, 2016
6. Michael Urban and Joel Murach, Python Programming, Shroff/Murach, 2016
7. Charles Severance. Python for Informatics: Exploring Information

B24CS1L02	OBJECT ORIENTED PROGRAMMING LAB	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

The aim of the course is to provide hands-on experience to the learners on various object oriented concepts in Java Programming. This course helps the learners to enhance the capability to design and implement various Java applications for real world problems.

Prerequisite

Topics covered under C Programming

Course Outcomes

After the completion of the course the student will be able to

CO 1	Identify the fundamentals of client/server systems, server types, characteristics, and the driving forces behind their development. (Cognitive Knowledge Level: Understand)
CO 2	Explain client and server components, hardware platforms, operating systems, and communication using the OSI model. (Cognitive Knowledge Level: Understand)
CO 3	Apply middleware concepts, RPC mechanisms, and ODBC architecture to select appropriate connectivity solutions for distributed client/server applications. (Cognitive Knowledge Level: Apply)
CO 4	Apply system management techniques such as remote administration tools, performance monitoring, and scalability strategies to maintain efficient client/server systems. (Cognitive Knowledge Level: Apply)
CO 5	Apply client/server database architectures and middleware components to design suitable database-driven client/server solutions. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	3	3	2	2	1	2	2	2	1	2
CO 2	2	3	2	2	1	2	2	2	1	2	1
CO 3	2	3	3	2	1	3	3	2	1	2	2
CO 4	2	3	3	3	3	3	3	3	3	3	2
CO 5	2	3	3	2	3	3	1	3	3	3	3

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Evaluation	20 marks
- Viva (Self Study)	10 marks
- Output	10 marks
Lab Record	5 marks
Viva-voce/Test	15 marks

SYLLABUS

LIST OF EXPERIMENTS

1	Write a Java program to check whether a given number is prime or not.
2	Write a Java program to find the second smallest element in an array.

3	Write a Java program that checks whether a given string is a palindrome or not. Ex: MALAYALAM is a palindrome.
4	Write a Java Program to find the frequency of a given character in a string.
5	Write a Java program to multiply two given matrices.
6	Write a Java program which creates a class named 'Employee' having the following members: Name, Age, Phone number, Address, Salary. It also has a method named 'print-Salary()' which prints the salary of the Employee. Two classes 'Officer' and 'Manager' inherits the 'Employee' class. The 'Officer' and 'Manager' classes have data members 'specialization' and 'department' respectively. Now, assign name, age, phone number, address and salary to an officer and a manager by making an object of both of these classes and print the same. (Exercise to understand inheritance)
7	Write a java program to create an abstract class named Shape that contains an empty method named numberOfSides(). Provide three classes named Rectangle, Triangle and Hexagon such that each one of the classes extends the class Shape. Each one of the classes contains only the method numberOfSides() that shows the number of sides in the given geometrical structures. (Exercise to understand polymorphism).
8	Write a Java program using reader/writer class that displays the number of characters, lines and words in a text file.
9	Write a Java program that reads from a file and writes to a file by handling all file related exceptions. (Using FileInputStream and FileOutputStream classes)
10	Write a Java program that reads a line of integers, and then displays each integer, and the sum of all the integers (Use String Tokenizer class of java.util)
11	Write a Java program that shows the usage of try, catch, throws and finally.
12	Write a Java program that implements a multi-threaded program which has three threads. First thread generates a random integer every 1 second. If the value is even, second thread computes the square of the number and prints. If the value is odd the third thread will print the value of cube of the number.
13	Write a Java program that shows thread synchronization.
14	Write a Java program that works as a simple calculator. Arrange Buttons for digits and the + - * % operations properly. Add a text field to display the result. Handle any possible exceptions like divide by zero. Use Java Swing.

15	Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green. When a radio button is selected, the light is turned on, and only one light can be on at a time. No light is on when the program starts.
16	Write a Java program to display all records from a table using Java Database Connectivity (JDBC).

Reference Books

1. Herbert Schildt, Java: The Complete Reference, 8/e, Tata McGraw Hill, 2011.
2. Paul Deitel, Harvey Deitel, Java How to Program, Early Objects 11th Edition, Pearson, 2018.
3. Y. Daniel Liang, Introduction to Java Programming, 7/e, Pearson, 2013.
4. Nageswararao R., Core Java: An Integrated Approach, Dreamtech Press, 2008.
5. Flanagan D., Java in A Nutshell, 5/e, O'Reilly, 2005.
6. Barclay K., J. Savage, Object Oriented Design with UML and Java, Elsevier, 2004.
7. Sierra K., Head First Java, 2/e, O'Reilly, 2005.
8. Balagurusamy E., Programming JAVA a Primer, 5/e, McGraw Hill, 2014.

B24MC1T03	PROFESSIONAL COMMUNICATION AND ETHICS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	0	1	3	P/F	2024

Preamble

This course aims to provide the students with the vital skills needed to excel in listening, reading, writing, and speaking. Whether conveying technical ideas or non-technical information, mastering these communication elements is crucial for aspiring professionals. The goal is to equip students with the ability to comprehend and successfully articulate ideas while also honing their persuasive communication skills. The course also aims to create in students awareness on ethics and human values.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to

CO 1	Expand vocabulary and linguistic proficiency pertinent to the field of engineering (Cognitive Knowledge Level: Apply)
CO 2	Examine, comprehend, and succinctly describe a range of textual material. (Cognitive Knowledge Level: Apply)
CO 3	Produce clear, technically sound documents and presentations that follow all required conventions. (Cognitive Knowledge Level: Apply)
CO 4	Manifest acute ethical awareness and effectively apply ethical principles in practical engineering scenarios. (Cognitive Knowledge Level: Apply)
CO 5	Analyze and address global ethical issues, showcasing an understanding of their roles as ethical leaders and contributors to technological development. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1								3	3	1	2
CO 2								3	3	1	2
CO 3	2	2	2			3	3	2	2	2	2
CO 4	2	2	2			3	3	3	3	1	2
CO 5	2	2	2			3	3	3	3	1	3

Assessment Pattern

Bloom's Category	Continuous Assessment	End Semester Examination (% Marks)
	Test (% Marks)	
Remember	30	30
Understand	40	40
Apply	30	30
Analyse		
Evaluate		
Create		

Mark Distribution

Total Marks	CIE Marks	ESE Marks
100	50	50

Continuous Internal Evaluation Pattern

Attendance	10 marks
Self Study (Regular Assessment)	15 marks
Series test (one test, conducted for 50 marks and reduced to 25)	25 marks

Regular assessment

Project report presentation and technical presentation through PPT	4 marks
Listening Test	2 marks
Group discussion/mock job interview	4 marks
Resume submission	2 marks
Assignment/Case study	3 marks

End Semester Examination Pattern

Total Marks: 50, Time: 2 hours. There will be two parts: Part A and Part B. Part A contains 4 questions carrying 5 marks each. Part B contains one question from each module in two sets of which students should answer one from each set. Each question can have a maximum of 2 sub-divisions and carry 15 marks each.

SYLLABUS

MODULE 1 (9 hours)

Communication Process

Modes, Verbal and Non-Verbal Communication, Verbal Aptitude- Misspelled Words, synonyms, paraphrasing, sentence completion using appropriate words, subject-verb agreement, Reading-Strategies for Effective Reading, types, Listening-Active and Passive Listening, Barriers, Taking notes while listening Activity- Worksheets, Exercises, Synthesizing and deriving conclusions from technical articles videos, and podcasts

MODULE 2 (9 hours)

Professional discipline

Public Speaking- Technical Talks- Formal and Informal Letters- Emails- Resume Preparation, Video Profile- GD Vs Debate-Dynamics of Professional Presentation (Individual and Group)- Format of Report, Proposal and Minutes.
Activity- Public Speaking, Podcast preparation, Resume preparation, Video profile creation, Company profiling, Group discussion, Technical Proposal, Structured Flow Analysis using AI.

MODULE 3 (9 hours)

Fundamentals of Ethical Engineering

Introduction to Human Values - Morals, Ethics, and Integrity - Academic and Work Ethics - Service Learning and Civic Virtue - Respect, Peaceful Living, Caring, and Sharing - Values of Honesty, Courage, Cooperation, Commitment, Empathy, and Self-Confidence - Senses of Engineering Ethics - Moral Autonomy and Ethical Theories - Moral Issues and Dilemmas in Engineering.

MODULE 4 (9 hours)

Professional Responsibility in a Global Context

Engineering as Social Experimentation - Responsible Experimentation and Codes of Ethics - Customs, Religion, and their Role in Engineering Ethics - Collegiality, Loyalty, and Conflict Management - Confidentiality, Conflicts of Interest, and Occupational Crime - Rights and Responsibilities in Engineering - Global Ethical Issues: Multinational Corporations, Environmental Ethics, Business Ethics, and Computer Ethics - Engineers as Leaders, Expert Witnesses, and Contributors to Technological Development.

Text Books

1. Ashraf Rizvi, "Effective Technical Communication", 2nd Edition, McGraw Hill Education, 2017.
2. Meenakshi Raman and Sangeetha Sharma, "Technical Communication: Principles and Practice", 2nd Edition, Oxford University Press, 2011
3. M Govindarajan, S Natarajan and V S Senthil Kumar, Engineering Ethics, PHI Learning Private Ltd, New Delhi, 2012.
4. R S Naagarazan, A text book on professional ethics and human values, New age international (P) limited, New Delhi, 2006.

Reference Books

1. English for Engineers and Technologists (Combined edition, Vol. 1 and 2), Orient Blackswan 2010.
2. Training in Interpersonal Skills: Tips for Managing People at Work, Pearson Education, India, 6th edition, 2015.
3. Mike W Martin and Roland Schinzinger, Ethics in Engineering, 4th edition, Tata McGraw Hill Publishing Company Pvt Ltd, New Delhi, 2014.
4. Charles D Fleddermann, Engineering Ethics, Pearson Education/ Prentice Hall of India, New Jersey, 2004.

5. Charles E Harris, Michael S Protchard and Michael J Rabins, Engineering Ethics- Concepts and cases, Wadsworth Thompson Learning, United states,2005.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Communication Process		9 hours
1.1	Modes, Verbal and Non-Verbal Communication	1
1.2	Verbal Aptitude- Misspelled Words, synonyms, paraphrasing	1
1.3	Sentence completion using appropriate words, subject verb agreement	1
1.4	Reading-Strategies for Effective Reading, types.	1
1.5	Listening-Active and Passive Listening, Barriers, Taking notes while listening.	1
1.6	Activity- Public Speaking, Podcast preparation, Resume preparation, Video profile creation, Company profiling, Group discussion, Technical Proposal.	4
Module 2: Professional discipline		9 hours
2.1	Public Speaking- Technical Talks- Formal and Informal Letters	1
2.2	Emails- Resume Preparation, Video Profile, GD Vs Debate	1
2.3	Dynamics of Professional Presentation (Individual and Group).	1
2.4	Format of Report, Proposal and Minutes.	1
2.3	Activity- Public Speaking, Podcast preparation, Resume preparation, Video profile creation, Company profiling, Group discussion, Technical Proposal.	5
Module 3: Fundamentals of Ethical Engineering		9 hours

3.1	Introduction to Human Values - Morals, Ethics, and Integrity	1
3.2	Academic and Work Ethics - Service Learning and Civic Virtue - Respect, Peaceful Living, Caring, and Sharing.	2
3.3	Values of Honesty, Courage, Cooperation, Commitment, Empathy, and Self-Confidence.	2
3.4	Senses of Engineering Ethics - Moral Autonomy and Ethical Theories.	2
3.5	Moral Issues and Dilemmas in Engineering.	2
Module 4: Professional Responsibility in a Global Context		9 hours
4.1	Engineering as Social Experimentation - Responsible Experimentation and Codes of Ethics.	1
4.2	Customs, Religion, and their Role in Engineering Ethics - Collegiality, Loyalty, and Conflict Management	2
4.3	Confidentiality, Conflicts of Interest, and Occupational Crime.	1
4.4	Rights and Responsibilities in Engineering - Global Ethical Issues.	1
4.5	Multinational Corporations, Environmental Ethics, Business Ethics, and Computer Ethics.	2
4.6	Multinational Corporations, Environmental Engineers as Leaders, Expert Witnesses, and Contributors to Technological Development.	2

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. Analyze how different modes of communication impact the overall message.
2. Identify and use appropriate verbal communication skills in various contexts.
3. Identify different types of reading and apply suitable strategies accordingly.
4. Recognize and overcome barriers to effective listening.

Course Outcome 2 (CO 2):

1. Demonstrate confidence and competence in public speaking.
2. Compose well-structured written communications.
3. Participate effectively in group discussions and debates, showcasing critical thinking and communication skills.

Course Outcome 3 (CO 3):

1. Understand the format and structure of professional reports and proposals.
2. Summarize and organize information effectively in meeting minutes.
3. Adapt presentation style based on the context and audience.

Course Outcome 4 (CO 4):

1. Explain the role of professional ethics in technological development.
2. Explain the need for environmental ethics in engineering projects.
3. How civic virtue and integrity contribute to application of ethical principles.

Course Outcome 5 (CO 5):

1. Explain how ethical issues in the workplace affect the development of a company.
2. Show how occupational crimes are resolved by keeping the rights of employees.
3. Explain the necessity of code of conduct for digital ethics.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SECOND SEMESTER B.TECH DEGREE EXAMINATION, JUNE 2025

Course Code: B24MC1T03

Course Name: PROFESSIONAL COMMUNICATION AND ETHICS

Max. Marks: 50

Duration: 2 hours

PART A

Answer all questions. Each question carries 5 marks.

1. Find out which pair of words 'can be filled-up in the blanks in the sentence in the same sequence to make the sentence grammatically correct and meaningfully complete.
 - (a) He was not to done the exercise himself.
 - a) expected, be b) required, being c) needed, get d) supposed, have
 - (b) A committee has been.....to.....the transformation of the city into an international finance center.
 - a) Constituted, convert b) appointed, oversee c) inducted, change d) converged, evaluate
2. Highlight the differences between a group discussion (GD) and a debate.
3. Briefly explain morals, values, and ethics.
4. Provide an explanation on conflicts of interest with an example.

PART B

Answer any one question from each set. Each question carries 15 marks.

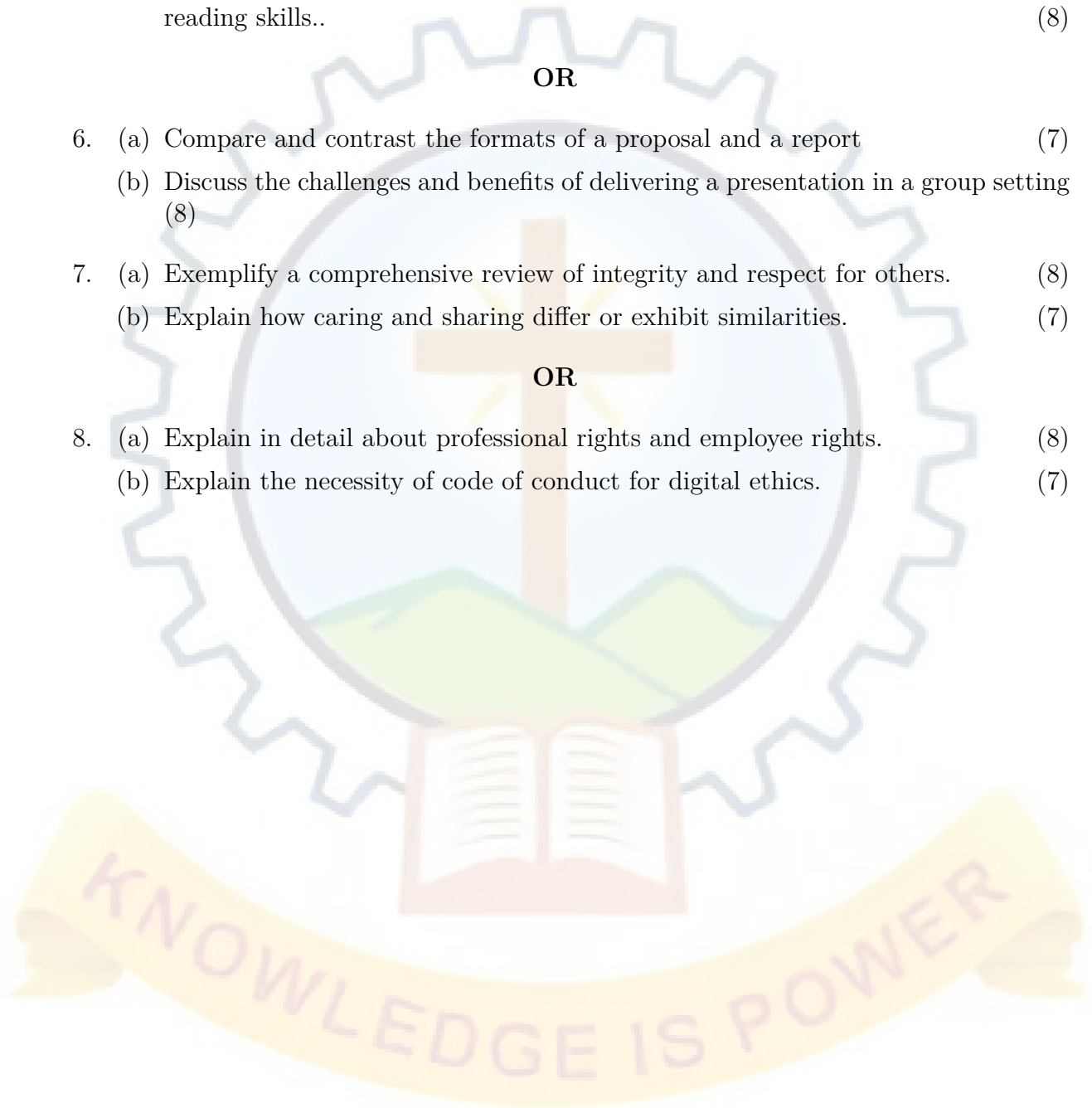
5. (a) “In today’s world, being a good listener is more important than being a good Speaker.” Enumerate (7)
(b) Help your friend by suggesting and explaining methods to improve his/her reading skills.. (8)

OR

6. (a) Compare and contrast the formats of a proposal and a report (7)
(b) Discuss the challenges and benefits of delivering a presentation in a group setting (8)
7. (a) Exemplify a comprehensive review of integrity and respect for others. (8)
(b) Explain how caring and sharing differ or exhibit similarities. (7)

OR

8. (a) Explain in detail about professional rights and employee rights. (8)
(b) Explain the necessity of code of conduct for digital ethics. (7)



B24MC1L02	IDEA LAB	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	2	2	P/F	2024

Preamble

This course enables the students to understand the concepts of design, development and documentation tools under various domains in engineering. The various topics covered in this course are concepts of 2D and 3D design, cutting, routing, engraving, milling, slicing, printing and fabrication. Students will be exposed to PCB design and populating. They will learn Microcontroller programming, embedded system design and technical documentation. This course helps students to analyse real-life problems and find solutions using multidisciplinary engineering.

Prerequisite

Higher secondary level Physics and Mathematics

Course Outcomes

After the completion of the course the student will be able to

CO 1	Create 2D and 3D models using appropriate tools. (Cognitive Knowledge Level: Analyze)
CO 2	Design and fabricate circuits using PCB Design and fabrication mechanisms. (Cognitive Knowledge Level: Analyze)
CO 3	Develop project using appropriate Micro controller Programming. (Cognitive Knowledge Level: Apply)
CO 4	Build a product for some applications using design and fabrication technologies. (Cognitive Knowledge Level: Create)
CO 5	Create electronic documentation for the system/project using appropriate tools. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes With Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1	3		2			1	1		
CO 2	1		1		1					1	1
CO 3	2	2	2	2	2	1	1	2	2	2	2
CO 4	1	2	3	2	3	3	3	3	3	3	3
CO 5						1			3		

Mark Distribution

Total Marks	CIE Marks	ESE (Internal) Project	Marks Micro	ESE (Internal) Project	Duration Micro Presentation
100	50	50			2 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Class Work/ Assessment	30 marks
Viva-Voce/ Test	10 marks

End Semester Evaluation Pattern:

Micro project Demonstration	20 marks
Micro Project Presentation	20 marks
Micro Project Report	10 marks

Note: The microproject has to be completed by the students as a group of a maximum of four students.

SYLLABUS

LIST OF EXPERIMENTS

Complete at least six experiments and one micro project from the given list.

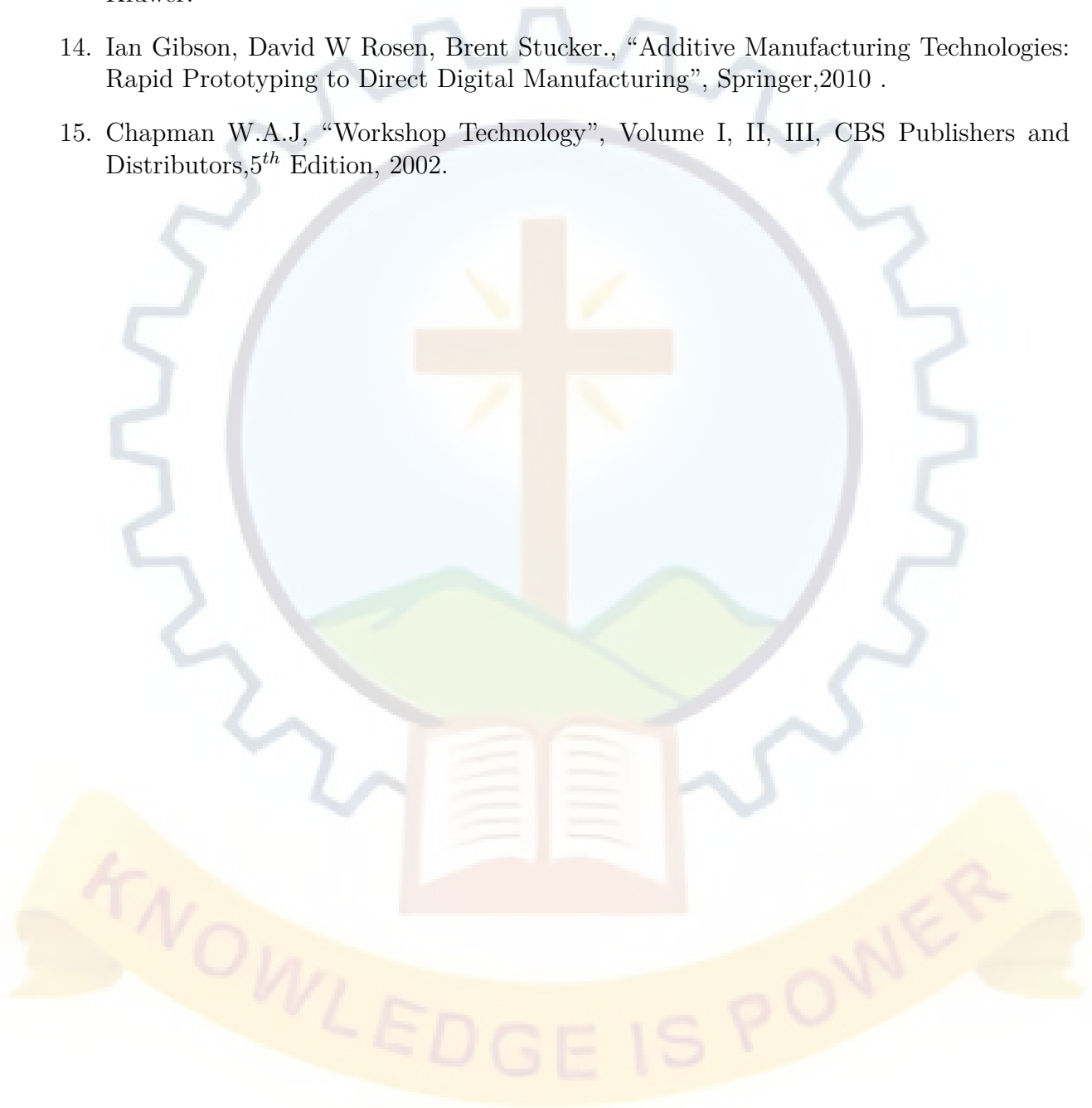
1	Prepare a 2D and 3D model using any standard tool.
2	Use the 2D model to engrave and cut the acrylic sheet using laser cutter. Assemble the laser-cut parts to fabricate the final model.
3	Use the 2D model for the fabrication of a model by using CNC milling.
4	Use a 3D model to engrave the pattern using CNC milling on the acrylic/wood/plastic block.

5	Use the 3D design for the fabrication of a model by using a 3D printer. Use a slicing software and generate the corresponding G-codes.
6	Write a program to read the input port pins of a micro controller and write the same to the output pins. Use a development board.
7	Write a program to read a sensor (temperature) and display it.
8	Write a program in Arduino IDE for Arduino development board to design a temperature controller. Control the speed of a fan based on the room temperature. Display the temperature on an LCD display.
9	Design a system to display the data send from the embedded system on a GUI in another Embedded system or PC (Wired – UART, I2C, SPI. Wireless – Bluetooth, Wifi)
10	Complete a Microproject. Prepare a technical report using latex for the temperature controller system in the standard template of the university.

Reference Books

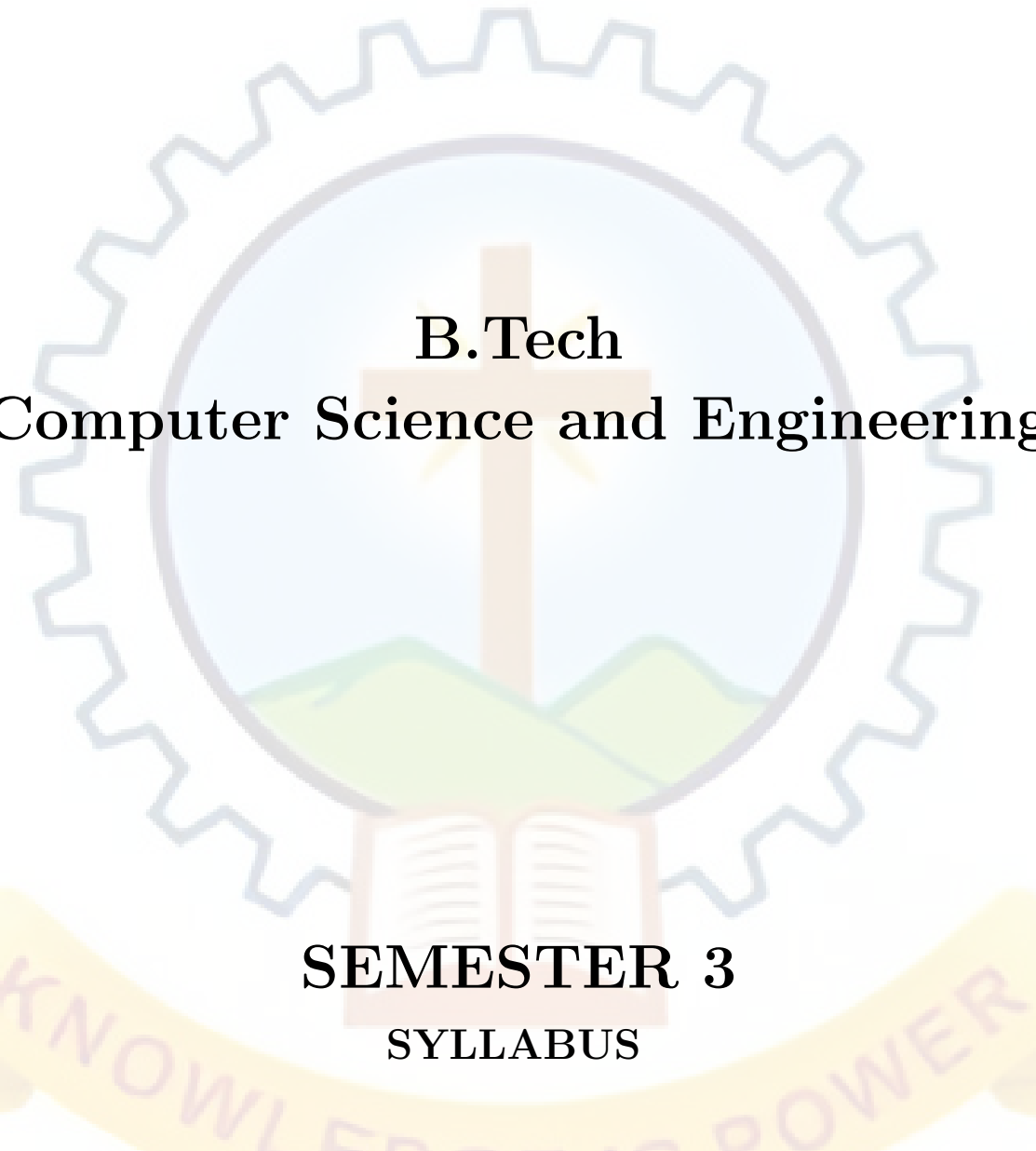
1. AICTE's Prescribed Textbook: Workshop / Manufacturing Practices (with Lab Manual), Khanna Book Publishing.
2. 3D Printing and Design, Dr. SabrieSoloman, ISBN: 978-9386173768, Khanna Book Publishing Company, New Delhi.
3. The Big Book of Maker Skills: Tools and Techniques for Building Great Tech Projects. Chris Hackett. Weldon Owen; 2018. ISBN-13: 978-1681884325.
4. The Total Inventors Manual (Popular Science): Transform Your Idea into a Top Selling Product. Sean Michael Ragan(Author).Weldon Owen;2017.ISBN-13:978-1681881584.
5. Make: Tools: How They Work and How to Use Them. Platt, Charles. Shroff/Maker Media. 2018. ISBN-13: 978- 352137374 .
6. The Art of Electronics. 3rd edition. Paul Horowitz and Winfield Hill. Cambridge University Press. ISBN: 9780521809269 .
7. Practical Electronics for Inventors. 4th edition. Paul Sherz and Simon Monk. McGraw Hill. ISBN-13: 978-1259587542 .
8. Encyclopedia of Electronic Components (Volume 1, 2 and 3). Charles Platt. Shroff Publishers. ISBN-13: 978-9352131945, 978-9352131952, 9789352133703.
9. Building Scientific Apparatus. 4th edition. John H. Moore, Christopher C. Davis, Michael A. Coplan and Sandra C. Greer. Cambridge University Press. ISBN-13: 978-0521878586.
10. Programming Arduino: Getting Started with Sketches. 2nd edition. Simon Monk. McGraw Hill. ISBN-13: 978-1259641633 .

11. Make Your Own PCBs with EAGLE: From Schematic Designs to Finished Boards. Simon Monk and Duncan Amos. McGraw Hill Education. ISBN-13: 978-1260019193.
12. Pro GIT. 2nd edition. Scott Chacon and Ben Straub. A press. ISBN-13: 9781484200773.
13. Venuvinod, PK., MA. W., Rapid Prototyping – Laser Based and Other Technologies, Kluwer.
14. Ian Gibson, David W Rosen, Brent Stucker., “Additive Manufacturing Technologies: Rapid Prototyping to Direct Digital Manufacturing”, Springer,2010 .
15. Chapman W.A.J, “Workshop Technology”, Volume I, II, III, CBS Publishers and Distributors,5th Edition, 2002.



MAR ATHANASIOUS COLLEGE OF ENGINEERING

Government Aided, Autonomous Institution
Kothamangalam, Kerala, India

The logo of Mar Athanasius College of Engineering is a circular emblem. It features a large gear-like outer border. Inside the circle, there is a light blue background with a central orange cross. Below the cross, there are green hills. At the bottom of the circle, there is an open book. A yellow banner with the text 'KNOWLEDGE IS POWER' in purple capital letters curves around the bottom of the emblem.

B.Tech Computer Science and Engineering

SEMESTER 3 SYLLABUS

SEMESTER 3

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24MA2T03C	DISCRETE MATHEMATICS	3-1-0-5	4	4
B	B24CS2T01	DATA STRUCTURES	3-1-0-5	4	4
C	B24CS2T02	COMPUTER ORGANIZATION AND ARCHITECTURE	3-1-0-5	4	4
D	B24CS2T03	COMPUTER NETWORKS	2-1-0-4	3	3
E	B24HU2T01	BUSINESS ECONOMICS AND FINANCIAL MANAGEMENT	3-0-0-4	3	3
G	B24CS2L03	DATA STRUCTURES LAB	0-0-3-2	3	2
H	B24CS2L04	NETWORKING LAB	0-0-3-2	3	2
I	B24MC2T04	UNIVERSAL HUMAN VALUE AND CONSTITUTIONAL RIGHTS	2-0-0-2	2	P/F
J	B24MC2T05	ENERGY CONSERVATION AND ENVIRONMENTAL SUSTAINABILITY	2-0-0-2	2	P/F
M	B24CSM3x	MINOR	3-1-0-5	4	4
		TOTAL		32	22

MINOR

COURSE NO.	COURSES
B24CSM31	PYTHON PROGRAMMING
B24CSM32	MATHEMATICS FOR MACHINE LEARNING
B24CSM33	DATA COMMUNICATION CONCEPTS

B24MA2T03C	DISCRETE MATHEMATICS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		2024

Preamble

The focus of the course is to build strong mathematical and logical reasoning skills that are essential for tackling more advanced topics in Computer Science. The course helps the learner to apply the theory and applications of elementary Counting Principles, Propositional Logic, Predicate Logic, Lattices, Generating Functions, recurrence relations, and Algebraic Structures eventually in practical applications

Prerequisites

A solid foundation in higher secondary school Mathematics.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Apply counting principles, permutations, combinations, the inclusion-exclusion principle, and derangements to solve combinatorial problems.(Cognitive Knowledge Level: Apply)
CO 2	Analyze recurrence relations and generating functions to model and solve discrete mathematical problems.(Cognitive Knowledge Level: Apply)
CO 3	Apply principles of mathematical logic, logical equivalences, quantifiers, and inference rules in formal reasoning and problem solving.(Cognitive Knowledge Level: Apply)
CO 4	Analyze relations, partial orders, equivalence relations, lattices, and their properties in discrete structures.(Cognitive Knowledge Level: Apply)
CO 5	Examine algebraic structures including semigroups, monoids, groups, homomorphisms, and isomorphisms and apply them to mathematical and computing contexts.(Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1							1		1
CO 2	2	1		1					1		1
CO 3	2	2	1						1		1
CO 4	2	2		1					1		1
CO 5	2	1	1	1					1		1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Counting Techniques:

The Rule of Sum – Extension of Sum Rule. The Rule of Product - Extension of Product Rule. The Binomial Theorem (without proof) Permutations. Combinations. Combination with Repetition. The Pigeonhole Principle. The Principle of Inclusion and Exclusion Theorem (without proof). Derangements.

MODULE 2 (9 hours)

Recurrence Relations and Functions:

Function – domain, range, one to one function, onto function, Image restriction. Generating Function - Definition and Examples, Calculation techniques, Exponential generating function. First-order linear recurrence relations with constant coefficients – homogeneous, non-homogeneous Solution. Second order linear recurrence relations with constant coefficients- homogeneous, non-homogeneous Solution.

MODULE 3 (9 hours)

Mathematical Logic:

Mathematical logic - Basic connectives and truth table, Statements, Logical Connectives, Tautology, Contradiction. Logical Equivalence - The Laws of Logic, The Principle of duality, Substitution Rules. The implication - The Contrapositive, The Converse, The Inverse. Logical Implication - Rules of Inference. The use of Quantifiers - Open Statement, Quantifier. Logically Equivalent – Contrapositive, Converse, Inverse, Logical equivalences and implications for quantified statement, Implications, Negation.

MODULE 4 (9 hours)

Relations:

Cartesian Product - Binary Relation. Properties of Relations- Reachability Relations, Reflexive Relations, Symmetric Relations, Transitive relations, Anti-symmetric Relations, Irreflexive relations Partial Order relations, Equivalence Relations. Partially ordered Set – Hasse Diagram, Maximal-Minimal Element, Greatest element, Least element, Least upper bound (lub), Greatest Lower bound(glb). Equivalence Relations and Partitions - Equivalence Class. Lattice, Sub lattice, Special Lattice, Complete Lattice, Bounded Lattice, Completed Lattice, Distributive Lattice.

MODULE 5 (9 hours)

Algebraic Structures:

Algebraic system-properties- Homomorphism and Isomorphism. Semi group and monoid, sub semi group and sub monoid, Homomorphism and Isomorphism of Semi group and monoids. Group- Elementary properties, subgroup, symmetric group on three symbols, Group Homomorphism, Isomorphism of groups, Cyclic group, Right Coset, Left Coset, Lagrange's Theorem.

Text Books

1. Discrete and Combinatorial Mathematics (An Applied Introduction), Ralph P Grimaldi, B. V Ramana, 8th Ramana, 5th Edition, Pearson.

Reference Books

1. Kenneth H. Rosen, Discrete Mathematics and Its Applications with Combinatorics and Graph Theory, Seventh Edition, MGH, 2011.
2. Tremblay J.P and Manohar R, "Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw-Hill Pub. Co. Ltd., New Delhi, 2003.
3. Bernard Kolman, Robert C. Busby, Sharan Cutler Ross, "Discrete Mathematical Structures", Pearson Education Pvt Ltd., New Delhi, 2003.
4. Richard Johnsonbaugh, "Discrete Mathematics", 5/e, Pearson Education Asia, New Delhi, 2002.
5. Joe L Mott, Abraham Kandel, Theodore P Baker, "Discrete Mathematics for Computer Scientists and Mathematicians", 2/e, Prentice-Hall India, 2009.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Counting Techniques		9 hours
1.1	The Rule of Sum – Extension of Sum Rule	1
1.2	The Rule of Product - Extension of Product Rule	1
1.3	The Binomial Theorem	1
1.4	Permutations	1

1.5	Combinations	1
1.6	Combination with Repetition	1
1.7	The Pigeonhole Principle	1
1.8	The Principle of Inclusion and Exclusion Theorem, Derangements.	2
Module 2: Recurrence Relations and Functions		9 hours
2.1	Function – domain, range, one to one function, onto function, Image restriction.	1
2.2	Generating Function - Definition and Examples, Calculation techniques.	2
2.3	Exponential generating function.	1
2.4	First order linear recurrence relations with constant coefficients – homogeneous.	1
2.5	First order linear recurrence relations with constant coefficients – non-homogeneous.	1
2.6	Second order linear recurrence relations with constant coefficients-homogeneous.	1
2.7	Second order linear recurrence relations with constant coefficients- non-homogeneous.	2
Module 3: Mathematical Logic		9 hours
3.1	Mathematical logic - Basic connectives and truth table.	1
3.2	Statements, Logical Connectives, Tautology, Contradiction.	1
3.3	Logical Equivalence - The Laws of Logic.	1
3.4	The Principle of duality, Substitution Rules, The implication - The Contrapositive, The Converse, The Inverse.	1
3.5	Logical Implication - Rules of Inference.	2
3.6	The use of Quantifiers - Open Statement, Quantifier. Logically Equivalent – Contrapositive, Converse, Inverse.	1
3.7	Logical equivalences and implications for quantified statement, Implications, Negation.	2
Module 4: Relations		9 hours
4.1	Cartesian Product - Binary Relation.	1

4.2	Properties of Relations- Reachability Relations, Reflexive Relations, Symmetric Relations, Transitive relations, Anti-symmetric Relations, Irreflexive relations.	1
4.3	Equivalence Relations.	1
4.4	Partial Order relations.	1
4.5	Partially ordered Set – Hasse Diagram.	1
4.6	Maximal element, Minimal Element, Greatest element, Least element, Least upper bound (lub), Greatest Lower bound(glb).	1
4.7	Equivalence Relations and Partitions - Equivalence Class.	1
4.8	Lattice, Sublattice.	1
4.9	Special Lattice-Complete Lattice, Bounded Lattice, Completed Lattice, Distributive Lattice.	1
Module 5: Algebraic Structures		9 hours
5.1	Algebraic system-properties- Homomorphism and Isomorphism.	1
5.2	Semi group and Monoid.	1
5.3	Sub semi group and Sub monoid, Homomorphism and Isomorphism of Semi group and monoids.	1
5.4	Group- Elementary properties, Subgroup.	1
5.5	Symmetric group on three symbols.	1
5.6	Group Homomorphism, Isomorphism of group.	1
5.7	Cyclic group.	1
5.8	Right Coset, Left Coset, Lagrange's Theorem.	2

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1): Solve counting problems by applying the elementary counting techniques- Rule of Sum, Rule of Product, Permutation, Combination, Binomial Theorem, Pigeon-hole Principle and Principle of Inclusion and Exclusion.

1. How many ways are there to place 12 identical marbles of the same colour and size in five distinct jars?
2. Find the number of integers between 1 and 1000 inclusive, which are not divisible by 5, 6 or 8.

3. How many cards must be selected from a deck of 52 cards to guarantee that at least three cards of the same suit are chosen.

Course Outcome 2 (CO 2): Explain Generating Functions and solve First Order and Second Order Linear Recurrence Relations with Constant Coefficients.

1. Solve $a_{n+2} - 2a_{n+1} + a_n = 2^n$.
2. A person invests Rs.100000 at 12% interest compounded annually. How long will it take to double the investment.
3. Find the generating function for the sequence $1, 3, 3^2, 3^3, \dots$.

Course Outcome 3 (CO 3): Check the validity of predicates in Propositional and Quantified Propositional Logic using truth tables, deductive reasoning and inference theory on Propositional Logic.

1. Negate the statement “Raj is doing his homework and Karen is practicing his piano lessons”.
2. Check whether the following arguments is valid. “If today is Tuesday then I have a test in Mathematics or Economics. If my Economics professor is sick then I will not have a test in Economics. Today is Tuesday and my Economics Professor is sick. Therefore, I have a test in Mathematics.”
3. Determine the truth value of the given statement where the domain consist of all real numbers.

$$\forall x \exists y (x < y)$$

Course Outcome 4 (CO 4): Classify binary relations into various types and illustrate an application of binary relation, Partially Ordered Sets and Lattices, in Computer Science.

1. If $A = \{1, 2, 3, 4\}$, give an example of a relation R that is reflexive and symmetric but not transitive.
2. Let Z be the set of integers. R is a relation called “Congruence Modulo 3” defined by $R = \{(x, y) / x \in Z, y \in Z, x - y \text{ is divisible by } 3\}$. Show that R is an equivalence relation.
3. Is $(D_{24}, |)$ a Bounded Lattice?

Course Outcome 5 (CO 5): Illustrate the abstract algebraic systems - Semigroups, Monoids, Groups, Homomorphism and Isomorphism of Monoids and Groups.

1. Give an example of a monoid which is not a group.
2. Show that every cyclic group is Abelian. Is the converse true?
3. Let H and K be subgroups of a group G , where e is the identity of G . Prove that if $|H| = 10$ and $|K| = 21$, then $H \cap K = \{e\}$.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM

THIRD SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2025

Course Code: B24MA2T03C

*Course Name: DISCRETE MATHEMATICS
Common to CSE, CSE(DS), AIML*

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Sixteen people are to be seated at two round tables, one of which seats ten people while the other seats 6. How many different seating arrangements are possible?
2. In how many ways can we distribute 8 identical white marbles in 4 distinct containers, so that no jar is left empty?
3. Suppose that the number of bacteria in a colony triples every hour. If 100 bacteria were used to begin the colony, how many bacteria will be there in the colony in 10 hours?
4. Solve the recurrence relation $a_n = 7a_{n-1}$, $n > 0$, $a_2 = 98$.
5. Write the negation of the following statement. "If I drive, then I will not walk".
6. Show that $(\neg p \vee \neg q) \rightarrow [p \wedge q \wedge r]$ and $p \wedge q$ are logically equivalent.
7. Let $A = \{1, 2, 3, 4, 5\}$ and $B = \{a, b, c, d\}$. How many elements are there in $P(A \times B)$, the power set of $A \times B$.
8. Define Complete Lattice. Give an example.
9. Is the set of integers Z , a semigroup under subtraction? Justify your answer.

10. Let $(A, \cdot)$ be a group. Show that $(ab)^{-1} = b^{-1}a^{-1}$.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) A certain ice cream store has 31 flavours of ice cream available. In how many ways can we order a dozen ice cream cones if, (7)
- i. We do not want the same flavour more than once.
 - ii. A flavour may be ordered as many as 12 times.
 - iii. A flavour may be ordered no more than 11 times.
- (b) Determine the number of six-digit integers (no leading zeros) in which (7)
- i. No digit may be repeated.
 - ii. Digits may be repeated.
 - iii. Digits may be repeated and the six-digit integer is even.
 - iv. No digit may be repeated and the six digit integer is divisible by 5.

OR

12. (a) In the expansion of $(x + 2y + 3z)^{12}$, determine the coefficient of (7)
- i. x^6y^6
 - ii. $x^5y^3z^4$
 - iii. x^4y^7z
- (b) In how many ways can the 26 letters of the alphabet be permuted so that none of the patterns car, dog, pun, or byte occurs? (7)
13. (a) Solve the recurrence relation $a_{n+2} - 4a_{n+1} + 3a_n = -200$, $n \geq 0$, $a_0 = 3000$, $a_1 = 3300$. (7)
- (b) In how many ways can a police officer distribute 24 rifle shells to four police officers so that each officer gets at least three shells but not more than eight. (7)

OR

14. (a) Solve the recurrence relation $a_{n+2} + a_n = 0$, $n \geq 0$, $a_0 = 0$, $a_1 = 3$. (7)
- (b) Determine the coefficient of x^{15} in the expansion of $(x^2 + x^3 + x^4 + \dots)^4$. (7)
15. (a) Prove the validity of the following argument:
“If it does not rain or if there is no traffic dislocation, then the sports day will be held and cultural programmes will go on”; “If the sports day is held, the trophy will be awarded” and “the trophy was not awarded” therefore, “It rained”. (7)
- (b) Write the symbolic form of the open statement ‘If x is even, then x is not divisible by 5’. Also write its converse, inverse and contrapositive both in symbolic form and in words. (7)

OR

16. (a) Write the dual of the statement $(\neg p \wedge \neg q) \vee (T_0 \wedge p) \vee p$ and use laws of logic to show that it is equivalent to $p \wedge \neg q$. (7)
- (b) Determine the truth value of each of the following statements if the universe comprises all non zero integers. (7)
- i. $\exists x \exists y (xy = 2)$
 - ii. $\exists x \forall y (xy = 2)$
 - iii. $\exists x \exists y [(4x + 2y = 3) \wedge (x - y) = 1]$
17. (a) Let $A = \{a, b, c\}$ and $Q(A)$, is the set of all proper subsets of A . The relation $\leq$ be the subset relation defined on $Q(A)$. Draw the Hasse diagram of $(Q(A), \leq)$. (7)
- (b) Let R be a relation defined on Z by if aRb if 4 divides $a - b$ Then find all equivalence classes. (7)

OR

18. (a) Let $A = \{1, 2, 3, 4, 6, 8, 12, 24\}$ and R be a relation on A defined by aRb if a divides b . Is R a partial order relation? (7)
- (b) For the partially ordered set $(\{3, 5, 9, 15, 24, 45\}, |)$. Find the (7)
- i. Minimal element
 - ii. Greatest element
 - iii. All upper bounds of $\{3, 5\}$
19. (a) Show that any group G is abelian if and only if $(ab)^2 = a^2b^2$ for all $a, b \in G$. (7)
- (b) Define semigroup homomorphism. Determine which of the following functions are homomorphism from $(Z^+, +)$ to $(Z^+, \cdot)$. (7)
- (i) $f(n) = n$
 - (ii) $f(n) = 3^n$

OR

20. (a) If $A = \{1, 2, 3\}$ list all permutations on A and prove that it is a group. (7)
- (b) State and prove Lagrange's theorem. (7)

B24CS2T01	DATA STRUCTURES	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		2024

Preamble

This course facilitates the learner to understand various data structures, their organization, and operations. The course introduces data structures such as stacks, queues, linked lists, binary trees, heaps, and graphs. The learner will be able to assess the most appropriate data structures and associated algorithms for tackling real world problems.

Prerequisites

Problem Solving and Programming Techniques(A) (B24ES1T01A)

Course Outcomes

After the completion of the course, the student will be able to:

CO 1	Develop algorithms to solve problems and evaluate their performance based of time/space complexities expressed using asymptotic notations. (Cognitive Knowledge Level: Apply)
CO 2	Solve problems using simple data structures like arrays and queues for applications involving arithmetic expressions, sparse matrices, polynomials, searching, etc. (Cognitive Knowledge Level: Apply)
CO 3	Construct solutions using linked lists for dynamic memory allocation and management. (Cognitive Knowledge Level: Apply)
CO 4	Use suitable non-linear data structures like trees and graphs to represent, store and manipulate hierarchical data efficiently. (Cognitive Knowledge Level: Apply)
CO 5	Make use of various sorting and hashing techniques to increase search efficiency and fast data retrieval. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	3	2	3	2				1		2
CO 2	3	3	2	3	3				3		2
CO 3	2	3	2	3	3				2		2
CO 4	2	3	3	3	3	2			3		3
CO 5	3	3	2	3	3	2			2		3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (5 hours)

Basic Concepts of Data Structures:

System Life Cycle, Algorithms, Performance Analysis, Space Complexity, Time Complexity, Asymptotic Notation, Complexity Calculation of Simple Algorithms.

MODULE 2 (10 hours)

Arrays and Searching:

Polynomial representation using Arrays, Sparse matrix, Stacks, Queues- Circular Queues, Priority Queues, Double Ended Queues, Evaluation of Expressions, Linear Search and Binary Search.

MODULE 3 (12 hours)

Linked List and Memory Management:

Self Referential Structures, Dynamic Memory Allocation, Singly Linked List- Operations on Linked List, Doubly Linked List, Circular Linked List, Stacks and Queues using Linked List, Polynomial representation using Linked List, Memory allocation and de-allocation, First-fit, Best-fit and Worst-fit allocation schemes.

MODULE 4 (8 hours)

Trees and Graphs:

Trees, Binary Trees- Tree Operations, Binary Tree Representation, Tree Traversals, Binary Search Trees- Binary Search Tree Operations. Graphs, Representation of Graphs, Depth First Search and Breadth First Search on Graphs, Applications of Graphs.

MODULE 5 (10 hours)

Sorting and Hashing:

Sorting Techniques – Selection Sort, Insertion Sort, Quick Sort, Merge Sort, and Heap Sort. Hashing- Hashing Techniques, Collision Resolution, Overflow handling, Hashing functions – Mid square, Division, Folding, Digit Analysis.

Text Books

1. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, Universities Press, Fundamentals of Data Structures in C.

Reference Books

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms, Fourth Edition, MIT Press.
2. Samanta D., Classic Data Structures, Prentice Hall India.
3. Richard F. Gilberg, Behrouz A. Forouzan, Data Structures: A Pseudocode Approach with C, 2/e, Cengage Learning.
4. Aho A. V., J. E. Hopcroft and J. D. Ullman, Data Structures and Algorithms, Pearson Publication.
5. Tremblay J. P. and P. G. Sorenson, Introduction to Data Structures with Applications, Tata McGraw Hill.
6. Wirth N., Algorithms + Data Structures = Programs, Prentice Hall.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Basic Concepts of Data Structures		5 hours
1.1	System Life Cycle,	1
1.2	Algorithms , Performance Analysis	1
1.3	Space Complexity, Time Complexity	1
1.4	Asymptotic Notation (Big O Notation)	1
1.5	Complexity Calculation of Simple Algorithms	1
Module 2: Arrays and Searching		10 hours
2.1	Polynomial representation using Arrays	1
2.2	Sparse matrix (Lecture 1)	1
2.3	Sparse matrix (Lecture 2)	1
2.4	Stacks	1
2.5	Queues, Circular Queues	1
2.6	Priority Queues,	1
2.7	Double Ended Queues,	1
2.8	Conversion and Evaluation of Expressions (Lecture 1)	1

2.9	Conversion and Evaluation of Expressions (Lecture 2)	1
2.10	Linear Search and Binary Search	1
Module 3: Linked List and Memory Management		12 hours
3.1	Self Referential Structures	1
3.2	Dynamic Memory Allocation	1
3.3	Singly Linked List-Operations on Linked List,	1
3.4	Doubly Linked List	1
3.5	Circular Linked List	1
3.6	Stacks Using Linked List	1
3.7	Queues Using Linked List	1
3.8	Polynomial representation using Linked List(Lecture 1)	1
3.9	Polynomial representation using Linked List(Lecture 2)	1
3.10	Memory de-allocation	1
3.11	Memory allocation-First-fit	1
3.12	Best-fit and Worst-fit allocation schemes	1
Module 4: Trees and Graphs		8 hours
4.1	Trees, Binary Trees	1
4.2	Tree Operations, Binary Tree Representation,	1
4.3	Tree Traversals	1
4.4	Binary Search Trees	1
4.5	Binary Search Tree Operations	1
4.6	Graphs, Representation of Graphs	1
4.7	Depth First Search and Breadth First Search on Graphs	1
4.8	Applications of Graphs	1
Module 5: Sorting and Hashing		10 hours
5.1	Sorting Techniques – Selection Sort	1
5.2	Insertion Sort	1
5.3	Quick Sort	1
5.4	Merge Sort	1
5.5	Heap Sort	1
5.6	Hashing- Hashing Techniques	1

5.7	Collision Resolution	1
5.8	Overflow handling	1
5.9	Hashing functions – Mid square and Division methods	1
5.10	Folding and Digit Analysis methods	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Write an algorithm for matrix multiplication and calculate its time complexity.
2. Explain the importance of time complexity in data structure. Derive the Big O notation for the function $f(n) = 4n^3 + 5n^2 + 7n + 3$.

Course Outcome 2 (CO 2):

1. How a linked list can be used to represent the polynomial $5x^4y^6 + 24x^3y^4 - 17x^2y^3 + 15xy^2 + 45$. Write an algorithm to add two Bivariate polynomials represented using linked list.
2. Write an algorithm to add two polynomials (single variable polynomials) represented using array.

Course Outcome 3 (CO 3):

1. Design a memory allocation system using a doubly linked list where each node represents a memory block (size in KB). Implement the following operations:
 - (a) Allocate memory (insert a node)
 - (b) Free memory (delete a node)
 - (c) Display the available memory blocksSimulate a memory allocation scenario where memory blocks (100KB, 200KB, 50KB, 300KB, 150KB) are allocated and then deallocated in a dynamic manner.
2. Consider six memory partitions of size 200 KB, 400 KB, 600 KB, 500 KB, 300 KB and 250 KB. These partitions need to be allocated to four processes of sizes 357 KB, 210 KB, 468 KB and 491 KB in that order. Perform the allocation of processes using-
 - (a) First Fit Algorithm
 - (b) Best Fit Algorithm
 - (c) Worst Fit Algorithm

Course Outcome 4 (CO 4):

1. Create a Binary Search Tree with node representing the following sequence 14, 15, 4, 18, 9, 16, 20, 17, 3, 7, 5, 2 and perform inorder, preorder and postorder traversals on the above tree and print the output.
2. Give an algorithm to perform the following operations in a single linked list.
 - (a) Count the number of nodes in a single linked list.
 - (b) Find the number of times a data item called ITEM occurs in the list.

Course Outcome 5 (CO 5):

1. The size of a hash table is 7. The index of the hash table varies from 0 to 6. Consider the keys 89, 18, 49, 58, 25 in the order. Show how the keys are stored in the hash table using Linear probing.
2. What is MAX HEAP? Write an algorithm to perform Heap sort for a set of values 16, 14, 10, 8, 7, 9, 3, 2, 4, 1.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

THIRD SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2025

Course Code: B24CS2T01

Course Name: DATA STRUCTURES

Max. Marks: 100

Duration: 3 hours

PART A

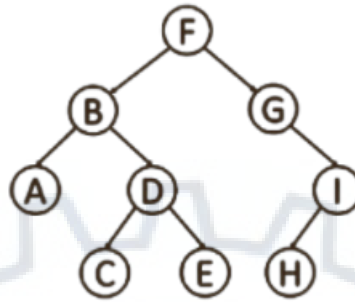
Answer all questions. Each question carries 3 marks.

1. Explain the best case, worst case, average case of linear search algorithm.
2. What is frequency count? Calculate the frequency count of the statement $x = x+1$; in the following code segment.

```
for (i = 0; i < n; i++)  
for (j = 1; j < n; j*=2)  
x = x + 1;
```

3. What are the applications of stack?
4. Differentiate between simple queue with circular queue.
5. Write an algorithm to insert an element at the end of doubly linked list. Illustrate with the help of an example.
6. What do you mean by self-referential structure? Give one example.
7. Differentiate between complete binary tree and full binary tree with suitable example.

8. Write the output of inorder, preorder, and postorder traversals on the following tree.



9. Illustrate insertion sort algorithm.

10. What do you mean by Double Hashing? Explain with an example.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain any three asymptotic notations used to express the complexity of algorithm with the help of suitable examples. (9)
(b) Derive Big-O notation for the function $f(n) = n^3 + 2n^2 + 5$. (5)

OR

12. (a) Write an algorithm to find the sum of n numbers and calculate its time complexity. (7)
(b) Explain various phases in system life cycle. (7)
13. (a) Write the algorithm to add two polynomials represented by arrays and illustrate with an example. (9)
(b) Write an algorithm to reverse a string using a stack. (5)

OR

14. (a) What do you mean by the data structure Priority Queue? Write algorithms to insert an element into a priority queue. (5)
(b) Write an algorithm to convert an infix expression into its equivalent postfix expression. Convert the expression $((A/(B-D+E))*(F-G)*H)$ to postfix form. Show each step in the conversion, including the stack contents. (9)
15. (a) What do you mean by a circular linked list? Write an algorithm to perform insert and delete operations on a circular linked list. (7)

- (b) Given five memory partitions of 400Kb, 600Kb, 350Kb, 200Kb, 800Kb (in order), how would the first-fit, best-fit, and worst-fit algorithms place processes of 520 Kb, 617 Kb, 200 Kb, and 750 Kb (in order)? (7)

OR

16. (a) Write an algorithm to insert an element at the beginning, end, and intermediate position of a doubly linked list. (7)
(b) Write an algorithm to append one linked list to another. Explain with an example. (7)
17. (a) Write and discuss algorithm to insert an element to Binary search tree. Show the structure of the binary search tree after adding each of the following values in that order: 2, 5, 1, 7, 10, 9, 11, 6. (7)
(b) Write and illustrate non-recursive pre-order tree traversal algorithm. (7)

OR

18. (a) Write and illustrate Depth First Search algorithm. (7)
(b) Explain various graph representations with example. (7)
19. (a) Write and illustrate Merge sort algorithm with a suitable example. (9)
(b) Write selection sort algorithm with the help of an example. (5)

OR

20. (a) Hash the following keys using open chaining method and closed linear probing method. Size of the table is 11, and the Hash function $H(K) = K \bmod 11$. Keys = 17, 22, 34, 23, 19, 66. (6)
(b) Explain any four hash functions with examples. (8)

B24CS2T02	COMPUTER ORGANIZATION AND ARCHITECTURE	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

The course is prepared with the view of enabling the learners capable of understanding the fundamental architecture of a digital computer. Study of Computer Organization and Architecture is essential to understand the hardware behind the code and its execution at physical level by interacting with existing memory and I/O structure. It helps the learners to understand the fundamentals about computer system design so that they can extend the features of computer organization to detect and solve problems occurring in computer architecture.

Prerequisites

Logic System Design (B24CS1T01)

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Comprehend the role of the basic components within a system and describe their importance in the overall functionality and performance of the system (Cognitive Knowledge: Apply)
CO 2	Examine the implementation aspects of arithmetic algorithms for floating-point numbers and provide an overview of pipeline processors, addressing key design issues that impact performance and efficiency. (Cognitive Knowledge Level: Apply)
CO 3	Explore different data transferring schemes, such as direct memory access (DMA), interrupt-driven I/O, and programmed I/O, within a system. Additionally, examine the principles of ALU design, focusing on the execution of arithmetic and logic operations within the processor (Cognitive Knowledge Level: Apply)
CO 4	Illustrate the principles of processor organization by designing various components, such as the Accumulator, status registers, and shifters and developing the control logic to efficiently solve a given arithmetic problem. (Cognitive Knowledge Level: Apply)
CO 5	Apply memory system principles, including RAM, ROM, and cache memory, to optimize performance through placement, replacement, and update policies. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	1	1		1	1		1		1
CO 2	2	3	2	3	3	3	1	1	2	2	3
CO 3	2	3	3	3	3	2	2	2	3	1	3
CO 4	2	3	3	3	3	2	2	2	3	1	3
CO 5	2	3	3	3	3	3	3	2	3	2	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignments)	Assignment/Analytical 5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Fundamentals of Computer Organization:

Basic Structure of Computers: Functional Units, Basic Operational Concepts - Complete Instruction Execution, Bus Structures- single bus organization and multiple bus organization, Memory System Basics: Memory Locations and Addresses, Memory Operations, Endianness, Instruction types and Classification, Addressing Modes based on RISC architecture.

MODULE 2 (10 hours)

Floating-Point Computation and Pipelined Processor Architecture:

Floating-Point Numbers and Operations- addition, subtraction, multiplication and division Operations on Floating-Point Numbers, Implementing Floating-Point Operations. Pipelined Processor: Pipelined Datapath, Pipelined Control, Hazards and Solutions: Data, Control and structural Hazards, hazards detection and resolution.

MODULE 3 (7 hours)

I/O Systems and ALU Design:

I/O Fundamentals: I/O Modules, I/O Processing Methods: Programmed I/O, Interrupt-Driven I/O, Direct Memory Access (DMA)-DMA controller, bus arbitration, Design of ALU (Arithmetic and Logic Unit)- Arithmetic Unit Design, Logic Unit Design, ALU design.

MODULE 4 (11 hours)

Processor Organization and Control Unit Design:

Accumulator register design, Shifter Design, status register, processor organization. Control logic design Design: hardwired control design(One Flip-Flop State), software control design (Micro programmed Control), microprogram sequencer-horizontal and vertical micro instruction.

MODULE 5 (8 hours)

Memory Architecture and Cache Optimization Techniques:

Introduction to Memory Systems: Memory Interleaving Techniques, Memory Organization, Semiconductor Memory RAM, ROM, and Types of cache memory: Basic Concepts, Cache Mapping Techniques, Cache Placement, Replacement and Update Policies.

Text Books

1. Computer Organization and Embedded Systems – V. Carl Hamacher, Zvonko G. Vranesic, Safwat G. Zaky, Naraig Manjikian, 6th Edition.
2. Computer Organization and Architecture: Designing for Performance – William Stallings, 10th Edition.
3. Computer System Architecture – M. Morris Mano, 3rd Edition.
4. Digital Design and Computer Architecture – David Money Harris, Sarah L. Harris, 2nd Edition.

Reference Books

1. Computer Architecture: A Quantitative Approach – John L. Hennessy, David A. Patterson, 6th Edition.
2. Structured Computer Organization – Andrew S. Tanenbaum, 6th Edition.
3. Fundamentals of Computer Organization and Architecture – Mostafa Abd-El-Barr, Hesham El-Rewini, 2nd Edition.
4. Modern Processor Design: Fundamentals of Superscalar Processors – John P. Shen, Mikko H. Lipasti, 1st Edition.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Fundamentals of Computer Organization		9 hours
1.1	Basic Structure of Computers: Functional Units	1
1.2	Basic Operational Concepts - Complete Instruction Execution	1

1.3	Bus Structures - Single Bus Organization	1
1.4	Bus Structures - Multiple Bus Organization	1
1.5	Memory System Basics: Memory Locations and Addresses	1
1.6	Memory Operations - Read and Write	1
1.7	Endianness: Big Endian vs. Little Endian	1
1.8	Instruction Types and Classification	1
1.9	Addressing Modes based on RISC Architecture	1
Module 2: Floating-Point Computation and Pipelined Processor Architecture		10 hours
2.1	Floating-Point Numbers and Operations: Implementing Floating-Point numbers	1
2.2	Addition and Subtraction	2
2.3	Multiplication Operations on Floating-Point Numbers	1
2.4	Division Operations on Floating-Point Numbers	1
2.5	Pipelined Processor	1
2.6	Pipelined Datapath	1
2.7	Pipelined Control	1
2.8	Data, Control and Structural Hazards	1
2.9	Hazards detection and resolution	1
Module 3: I/O Systems and ALU Design		7 hours
3.1	I/O Fundamentals: I/O Modules, I/O Processing Method	1
3.2	Programmed I/O, Interrupt-Driven I/O	1
3.3	Interrupt-Driven I/O	1
3.4	Direct Memory Access (DMA)	1
3.5	DMA controller and bus arbitration	1
3.6	Design of ALU (Arithmetic and Logic Unit) Arithmetic Unit Design	1
3.7	Logic Unit Design, ALU design	1
Module 4: Processor Organization and Control Unit Design		11 hours
4.1	Accumulator register design	1
4.2	Shifter Design	1
4.3	Status register	1

4.4	processor organization	1
4.5	Control logic design Design	1
4.6	Hardwired control design(one flip flop per state)	1
4.7	Hardwired control design(one flip flop per state)	1
4.8	Software control logic design(micro program control)	1
4.9	Software control logic design (micro program control)	1
4.10	Microprogram sequencer	1
4.11	Horizontal and Vertical micro instruction	1
Module 5: Memory Architecture and Cache Optimization Techniques		8 hours
5.1	Memory interleaving	1
5.2	Memory organization	1
5.3	Semiconductor RAM memories	1
5.4	ROM and types	1
5.5	Cache Memory: Basic Concepts	1
5.6	Cache Mapping Techniques	2
5.7	Cache Replacement Policies	1
5.8	Memory update Policies	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Identify the addressing modes that can be used for representing the following higher level language constructs in machine level. Illustrate each addressing mode using an example.
 - (a) Arrays
 - (b) Pointers
 - (c) Constants
 - (d) Variables
2. Illustrate the single bus organization of processor unit with the help of suitable diagrams. Explain, listing the control signals, how the following operations are handled in this organization.
 - (a) Move (R6), R2

(b) Add (R4),(R3)

Course Outcome 2 (CO 2):

1. Identify the various hazard that might occur during the execution of the following instruction i,j,k in a pipeline structure which consists of five stages of instruction, fetch, decode, operation fetch, execute and write?

SUB R2, R1, R3
OR R3, R2, [R5]
JUMP LOOP
LOOP: SUB R2, R1, R3
SUB R2, R1, R3
JCLOC

2. A computer system using IEEE 754 single-precision floating-point representation performs the addition operations on : $3.5 \times 10^5 + 7.5 \times 10^{-4}$.

Course Outcome 3 (CO 3):

1. Design an arithmetic circuit with one selection variable S and two data inputs A and B. When S=0, the circuit performs the addition operation $F=A+B$, when S=1 the circuit perform increment operation $F=A+1$.
2. Describe the different I/O processing methods and their impact on system performance.

Course Outcome 4 (CO 4):

1. Design an accumulator with control variables P_1 to P_5 , which perform the following arithmetic and logic operations:
 - (a) $P_3 : A \leftarrow A \wedge B$ (Bitwise XOR)
 - (b) $P_4 : A \leftarrow A \oplus B$ (Bitwise AND)
 - (c) $P_5 : A \leftarrow \text{Shl}(A)$ (Logical Shift Right)
2. Compare one flip-flop state control and microprogrammed control in processor design.

Course Outcome 5 (CO 5):

1. How do multi-level caches improve system performance? Provide examples of real-world implementations.
2. Compare Asynchronous DRAMs and Synchronous DRAMs.

MODEL QUESTION PAPER

QP CODE:

Pages: 4

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

THIRD SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2025

Course Code: B24CS2T02

Course Name: Computer Organization and Architecture

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Auto increment mode is useful for accessing data items in successive memory locations. Justify the statement.
2. Consider a system with 32bit processor, the program reads the word “BEAUTIFUL”, and store it in successive memory locations starting from location 1000. Show the contents of the second and third word in the memory locations when the word is stored in the following format.
 - (a) Big-endian format
 - (b) Little-endian format

The ASCII values are as follows:

- B = 0x42
- E = 0x45
- A = 0x41
- U = 0x55
- T = 0x54
- I = 0x49
- F = 0x46

- $U = 0x55$
- $L = 0x4C$

3. What is IEEE 754 floating-point representation?
4. How does pipelining improves performance?
5. Differentiate between programmed I/O and interrupt-driven I/O.
6. Consider the designing of Arithmetic circuit. The arithmetic operations can be performed by keeping A as such and making changes to the B value. Design a circuit to accomplish this with the help of a neat diagram.
7. What is the function of a status register in a processor?
8. Differentiate between hardwired and microprogrammed control units.
9. What is the purpose of cache memory in a computer system?
10. Differentiate between direct-mapped and associative cache memory.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Identify the addressing modes that can be used for representing the following higher level language constructs in machine level. Illustrate each addressing mode using an example. (7)
 - i. Arrays
 - ii. Pointers
 - iii. Constants
 - iv. Variables
- (b) Illustrate the single bus organization of processor unit with the help of suitable diagrams. Explain, listing the control signals, how the following operations are handled in this organization. (7)
 - i. Transfer contents of register R5 to R1
 - ii. Move (R6), R2

OR

12. (a) Explain memory addressing in a computer. How does addressing affect system performance? (7)
- (b) Discuss the various addressing modes used in RISC architecture with suitable examples. (7)
13. (a) Explain the steps involved in floating-point addition and subtraction. How are rounding errors handled? (7)

- (b) Identify the various hazard that might occur during the execution of the following instruction i,j,k in a pipeline structure which consist of five stages of instruction, fetch, decode, operation fetch, execute and write? (7)

```
SUB R2, R1, R3
OR R3, R2, [R5]
JUMP LOOP
LOOP: SUB R2, R1, R3
SUB R2, R1, R3
JCLOC
```

OR

14. (a) A computer system using **IEEE 754 single-precision floating-point representation** performs the following operations: (7)
- A. Addition: $1.5 \times 10^3 + 2.5 \times 10^{-2}$
 - B. Multiplication: $3.75 \times 10^4 \times 2.0 \times 10^{-3}$
 - i. Identify any potential precision loss or rounding errors in the operations.
 - ii. Explain how the system handles normalization during these calculations.
- (b) Analyze how performance is measured in pipelined processors. Use an example to illustrate your answer. (7)
15. (a) Explain the working of direct memory access (DMA) and its advantages over traditional I/O methods. (7)
- (b) Design an arithmetic circuit with one selection variable S and two data inputs A and B. When S=0, the circuit performs the addition operation $F=A+B$, when S=1 the circuit perform increment operation $F=A+1$. (7)

OR

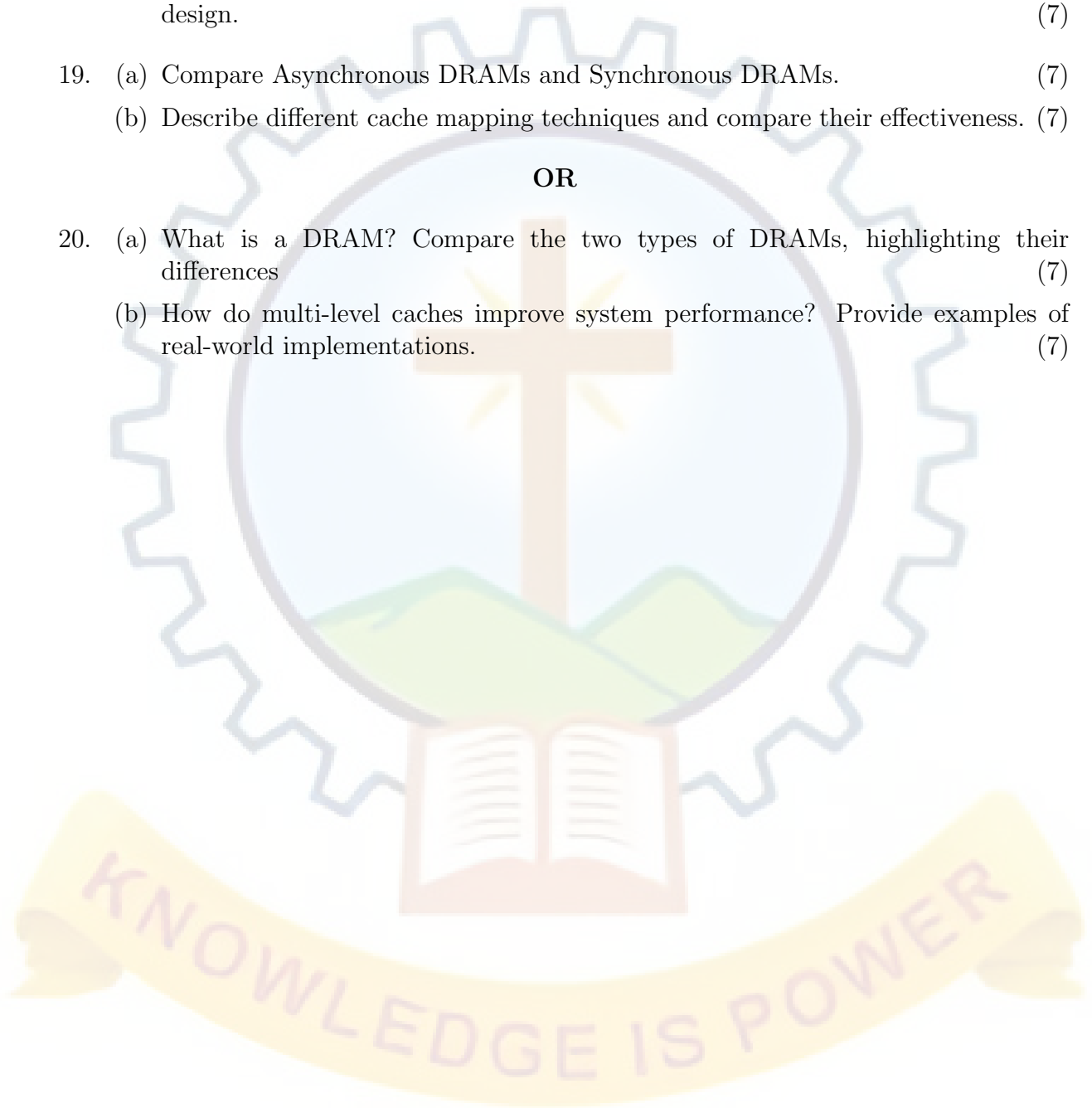
16. (a) Describe the different I/O processing methods and their impact on system performance. (7)
- (b) Explain how ALU operations are performed using logic gates and discuss the role of status registers in arithmetic computations. (7)
17. (a) Design an accumulator with control variables P_1 to P_5 , which perform the following arithmetic and logic operations: (7)
- i. $P_1 : A \leftarrow A + B$ (Addition)
 - ii. $P_2 : A \leftarrow A'$ (One's Complement)
 - iii. $P_3 : A \leftarrow A \wedge B$ (Bitwise AND)
 - iv. $P_4 : A \leftarrow A \oplus B$ (Bitwise XOR)
 - v. $P_5 : A \leftarrow \text{Shl}(A)$ (Logical Shift Left)
- (b) Explain the working of a control unit. How does it influence instruction execution? (7)

OR

18. (a) Design a 4-bit combinational logic shifter which will perform the operations given below with two control variable H1 and H0? (7)
- i. ashr
 - ii. Clear
 - iii. Load all bits with 1.
- (b) Compare one flip-flop state control and microprogrammed control in processor design. (7)
19. (a) Compare Asynchronous DRAMs and Synchronous DRAMs. (7)
- (b) Describe different cache mapping techniques and compare their effectiveness. (7)

OR

20. (a) What is a DRAM? Compare the two types of DRAMs, highlighting their differences (7)
- (b) How do multi-level caches improve system performance? Provide examples of real-world implementations. (7)



B24CS2T03	COMPUTER NETWORKS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble This course introduces students to a clear understanding of how computer networks from local area networks to the massive and global Internet are built, how they allow computers to share information and communicate with one another. This course explores the physical aspects of computer networks, the layers of the OSI Reference Model, and inter-networking. It enables learners to compare and analyze existing network technologies, helping them select the most suitable network design for a given system.

Prerequisites

Nil

Course Outcomes

After the completion of the course, the student will be able to:

CO 1	Understand the fundamental concepts, types, and applications of computer networks, including computer hardware, software, and network hardware, software, and layered architecture models.(Cognitive Knowledge Level: Understand)
CO 2	Analyze data link layer functionalities, error detection and correction techniques, and medium access control protocols for efficient data transmission.(Cognitive Knowledge Level: Apply)
CO 3	Demonstrate an understanding of network layer operations, including addressing, routing algorithms, and congestion control mechanisms. (Cognitive Knowledge Level: Apply)
CO 4	Examine transport layer services, protocols, flow control, congestion control, and NAT mechanisms.(Cognitive Knowledge Level: Apply)
CO 5	Evaluate application layer protocols, security challenges, and cryptographic techniques to enhance network security and performance.(Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

Assessment Pattern

Mark Distribution

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1				1							1
CO 2	2	2	1	1							1
CO 3	2	1	1								1
CO 4	2	1	1								1
CO 5		1					2				1

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyse			
Evaluate			
Create			

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction & Physical Layer

Introduction to Computer Networks, Network Hardware and Software, OSI Model: Overview and Functions of Each Layer, TCP/IP Model: Comparison with OSI Model, Physical Layer: Transmission Modes, Signal Encoding Techniques.

MODULE 2 (8 hours)

Data Link Layer

Data Link Layer: Functions and Services, Error Detection and Correction: CRC, Checksum, Flow Control Mechanisms: Stop-and-Wait, Sliding Window Protocols, Medium Access Control (MAC) Protocols: ALOHA, CSMA/CD, CSMA/CA, IEEE 802.3 Ethernet (Wired LAN) and IEEE 802.11 (Wireless LAN).

Case Study: Wireshark Analysis of Ethernet & Wireless LAN.

MODULE 3 (8 hours)

Network Layer

Network Layer Services and Functions, IPv4 Addressing and Subnetting, IPv6 Addressing and Transition Strategies, Routing Algorithms: Shortest Path, Distance Vector, Link State Routing Protocols: RIP, OSPF, BGP, Congestion Control Techniques - Leaky Bucket, Token Bucket, Quality of Service (QoS) Techniques for Network Performance.

MODULE 4 (8 hours)

Transport Layer

Transport Layer Services & Protocols, User Datagram Protocol (UDP) – Features and Applications, Transmission Control Protocol (TCP) – Features, Segment Structure, TCP Connection Establishment and Flow Control TCP Congestion Control Mechanisms - Slow Start, AIMD, Fast Retransmit.

Case Study: Analyzing TCP and UDP Packets using Wireshark.

MODULE 5 (5 hours)

Application Layer and Network Security

Overview of Application Layer Protocols, Domain Name System (DNS) and its Role in

the Internet, File Transfer Protocol (FTP) and Simple Mail Transfer Protocol (SMTP), Hypertext Transfer Protocol (HTTP) and World Wide Web (WWW).

Case Study: Cybersecurity Attacks and Network Defense Strategies

Text Books

1. Behrouz A Forouzan, Data Communication and Networking, 4/e, Tata McGraw Hill
2. Andrew S. Tanenbaum, Computer Networks, 4/e, PHI (Prentice Hall India).

Reference Books

1. Larry L Peterson and Bruce S Dave, Computer Networks – A Systems Approach, 5/e, Morgan Kaufmann
2. Fred Halsall, Computer Networking and the Internet, 5/e.
3. James F. Kurose, Keith W. Ross, Computer Networking: A Top-Down Approach, 6/e.
4. Keshav, An Engineering Approach to Computer Networks, Addison Wesley, 1998.
5. W. Richard Stevens. TCP/IP Illustrated Volume 1, Addison-Wesley, 2005.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction & Physical Layer		7 hours
1.1	Introduction to Computer Networks: Uses, Types, and Applications	1
1.2	Network Hardware and Software: LAN, WAN, MAN, Internet, Protocols	1
1.3	OSI Model: Overview and Functions of Each Layer	1
1.4	TCP/IP Model: Comparison with OSI Model	1
1.5	Physical Layer: Transmission Modes, Signal Encoding Techniques	1

1.6	Transmission Media: Guided (Twisted Pair, Coaxial, Fiber Optic)	1
1.7	Transmission Media Unguided (Radio, Infrared, Microwave)	1
Module 2: Data Link Layer		8 hours
2.1	Data Link Layer: Functions and Services	1
2.2	Error Detection & Correction: CRC, Checksum, Hamming Code	1
2.3	Flow Control Mechanisms: Stop-and-Wait, Sliding Window Protocols	1
2.4	Medium Access Control (MAC) Protocols: ALOHA, CSMA/CD	1
2.5	Medium Access Control (MAC) Protocols: CSMA/CA	1
2.6	IEEE 802.3 Ethernet (Wired LAN)	1
2.7	IEEE 802.11 (Wireless LAN)	1
Case Study: Wireshark Analysis of Ethernet & Wireless LAN (1 hour)		
Module 3: Network Layer		8 hours
3.1	Network Layer Services & Functions	1
3.2	IPv4 Addressing and Subnetting	1
3.3	IPv6 Addressing and Transition Strategies	1
3.4	Routing Algorithms: Shortest Path, Distance Vector	1
3.5	Routing Algorithms: Link state Routing	1
3.6	Routing Protocols: RIP, OSPF, BGP	1
3.7	Congestion Control Techniques (Leaky Bucket, Token Bucket)	1
3.8	Quality of Service (QoS) Techniques for Network Performance	1
Module 4: Transport Layer		8 hours
4.1	Transport Layer Services & Protocols	1
4.2	User Datagram Protocol (UDP) – Features and Applications	1
4.3	Transmission Control Protocol (TCP) – Features, Segment Structure	1
4.4	TCP Connection Establishment	1

4.5	Flow Control	1
4.6	TCP Congestion Control Mechanisms (Slow Start, AIMD, Fast Retransmit)	1
4.7	TCP Congestion Control Mechanisms Continued.	1
Case Study: Analyzing TCP and UDP Packets using Wireshark (1 hour)		
Module 5: Application Layer and Network Security		5 hours
5.1	Overview of Application Layer Protocols	1
5.2	Domain Name System (DNS) and its Role in the Internet	1
5.3	File Transfer Protocol (FTP) and Simple Mail Transfer Protocol (SMTP)	1
5.4	Hypertext Transfer Protocol (HTTP) and World Wide Web (WWW)	1
Case Study: Cybersecurity Attacks and Network Defense Strategies(1 hour)		

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Compare TCP/IP and OSI reference model.
2. The purpose of physical layer is to transport a raw bit stream from one machine to another. Justify.

Course Outcome 2 (CO 2):

1. Ethernet frames must be at least 64 bytes long to ensure that the transmitter is still going in the event of a collision at the far end of the cable. Fast Ethernet has the same 64-byte minimum frame size but can get the bits out ten times faster. How is it possible to maintain the same minimum frame size?
2. What do you mean by bit stuffing?

Course Outcome 3 (CO 3):

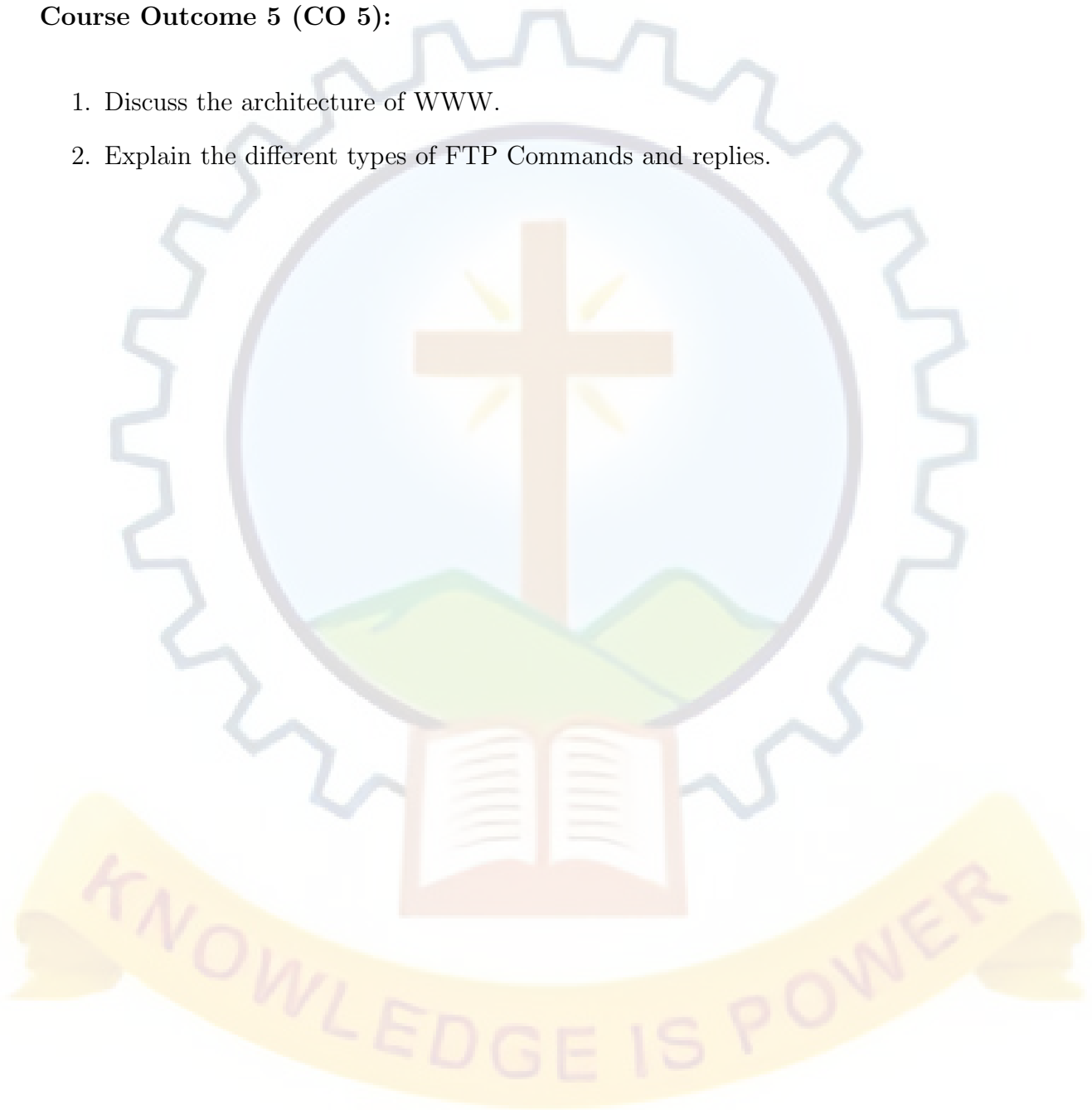
1. Discuss the differences between IPV4 and IPV6.
2. Explain the different types of routing algorithms.

Course Outcome 4 (CO 4):

1. Discuss the differences between TCP and UDP.
2. Enumerate the different congestion control mechanisms in TCP.

Course Outcome 5 (CO 5):

1. Discuss the architecture of WWW.
2. Explain the different types of FTP Commands and replies.



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

THIRD SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2025

Course Code: B24CS2T03

Course Name: COMPUTER NETWORKS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. What does “negotiation” mean when discussing network protocols in a layered architecture? Give an example.
2. Define simplex, half-duplex, and full-duplex transmission modes. Give one example for each.
3. Data link protocols almost always put the CRC in a trailer rather than in a header. Why?
4. An 8-bit byte with binary value 10101111 is to be encoded using an even-parity Hamming code. What is the binary value after encoding?
5. Illustrate the Count to Infinity problem in routing.
6. Discuss the IPV6 frame format.
7. Explain the TCP Segment Header.
8. Why is Transport layer called true End to End layer? Why is flow control and error control used in Transport layer in addition to data link layer ?
9. Discuss the ports used in FTP?
10. What is SMTP.Explain its features.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Discuss the ISO-OSI model with the help of a neat diagram. (14)

OR

12. (a) Explain the different signal encoding techniques in the physical layer. (8)
(b) Compare the TCP/IP layer with the ISO-OSI layer in the physical layer. (8)
13. (a) Discuss the GoBackN and selective repeat sliding window protocols. (8)
(b) Differentiate IEEE 802.3 with IEEE 802.11. (6)

OR

14. (a) Differentiate the different Medium Access Control(MAC) protocols. (14)
15. (a) Explain the different types of routing algorithms. (14)

OR

16. (a) What are the techniques for ensuring good QoS? (8)
(b) Explain the Leaky Bucket Algorithm. (6)
17. (a) A company is granted the site address 181.56.0.0 (class B). The company needs 1000 subnets. Find the number of possible subnets and the hosts that can be connected to each subnet. (8)
(b) What are the TCP Congestion control mechanisms. (6)

OR

18. (a) What are the differences between TCP and UDP. (8)
(b) Describe the structure of UDP Header. (6)
19. (a) Discuss DNS and its role in Internet? (8)
(b) Explain the different types of FTP Queries and Replies? (6)

OR

20. (a) Discuss the features of SMTP? (8)
(b) Explain the features of WWW? (6)

B24HU2T01	BUSINESS ECONOMICS AND FINANCIAL MANAGEMENT	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	0	0	4	3	2024

Preamble

This course will aid and equip the students to comprehend the various concepts in Business Economics and Finance. They will gain an understanding of price, demand, production, costs and revenue. They will also learn about the functioning of various markets and fathom the problems affecting the world of business. They will be introduced to national income accounting and to the financial tools used in personal finance. The students will also gain an insight into business financing and the functioning of the stock market.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO1	Understand the fundamental concepts and theories of demand, supply and production in relation to various economic issues pertaining to the engineering industry. (Cognitive Knowledge Level: Understand)
CO2	Apply the concepts relating to costs and revenue in the functioning of engineering firms in different market situations and solve industry level business problems using tools like break even analysis. (Cognitive Knowledge Level: Apply)
CO3	Apply the fundamental macroeconomic principles to emerging economic concepts influencing the growth and performance of the economy as a whole like national income accounting, monetary and fiscal policy, balance of payments and international trade. (Cognitive Knowledge Level: Apply)
CO4	Apply the fundamental concepts of financial management to acquire knowledge in the functioning of the Indian financial system and evaluate decisions regarding personal finance. (Cognitive Knowledge Level: Apply)
CO5	Demonstrate decision-making capability by acquiring knowledge in industrial financing, international financing, stock markets and mutual funds. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	1	1		1					1	1	1
CO 2	1	1	1	1	1	1			1	1	1
CO 3	1	1		1	1				1	1	1
CO 4	1		1	1		1	1	1	1	2	1
CO 5	1	1	1			1	1	1	1	1	1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	50	30	30
Understand	50	40	40
Apply		30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (8 hours)

Fundamentals of Business Economics

Introduction – Demand and its determinants – Law of demand – Elasticity of demand: Price, Income and Cross - Measurement of elasticity and its applications (Numerical problems) – Supply and its determinants – Determination of Equilibrium Price – Changes in demand and supply and its effects Utility – Law of diminishing marginal utility - Consumer surplus - Producer surplus. Production concepts: Production function - Cobb Douglas function (Numerical problems) - Average product - Marginal product - Law of variable proportions – Law of Returns to Scale.

MODULE 2 (7 hours)

Cost, Revenue and Markets

Concepts of cost: Opportunity cost - Explicit and implicit cost – Private and social cost- Short run cost curves – Fixed, variable, total, average and marginal cost curves - Long run cost curves.

Concepts of revenue: Average and marginal revenue - Shutdown point - Break Even analysis (Numerical problems).

Markets: Perfect Competition, Monopoly, Monopolistic Competition, Oligopoly - Cartel and Collusion (Features and equilibrium of a firm) - Product pricing: Cost plus pricing – Target return pricing – Penetration pricing – Predatory pricing – Going rate pricing – Price skimming – Administered pricing.

MODULE 3 (7 hours)

National Income, Inflation and International Trade

Circular flow of income - Multi-sector model - National income concepts: GNP, GDP, NNP, NI, PI, DPI, PCI - Methods of measuring national income – Difficulties (Numerical problems).

Inflation –Types - Causes and effects – Measures to control inflation - Monetary and fiscal policies – Deflation.

International Trade - Balance of payments – Components – Deficit - Devaluation – Tariff and non-tariff barriers.

MODULE 4 (7 hours)

Fundamentals of Financial Management

Introduction - Reserve Bank of India – Functions - Credit control techniques: Quantitative and qualitative techniques - Working capital management - Factors affecting working capital - Management of cash and marketable securities - Receivables management - Balance Sheet - Profit and Loss Account.

Personal Finance: Personal budget – Tracking income and expenses - 50-30-20 budgeting rule – Emergency fund – Debit vs Credit instruments – Diversification of Investments – Shares vs Bonds - Power of Compounding – Financial independence – Types of Insurance - Digital technology in Finance.

MODULE 5 (7 hours)

Business Financing

Introduction: The Stock Market: Functions, Problems faced by the stock market in India – Demat account and trading account – Market indices: Sensex and Nifty - Derivatives: Forwards, Futures, Options, Swaps - Mutual Funds – Types.

Sources of business financing: Equity capital - Preference capital - Debenture capital - Term loans - Retained earnings - Money market – Instruments - International Financing - FDI, FII.

Text Books

1. Dominic Salvatore “Principles of Microeconomics”, Oxford University Press, 2009.
2. Gregory N Mankiw, “Principles of Macro Economics”, Cengage Learning India, 2022.
3. Prasanna Chandra, “Financial Management”, McGraw Hill, 2022.

Reference Books

1. Paul A Samuelson, “Economics”, McGraw Hill, 2019.
2. A. Koutsoyiannis, “Modern microeconomics”, Palgrave MacMillan, 1979.
3. Geetika Piyali Ghosh and Chodhury “Managerial Economics”, McGraw Hill, 2017.
4. M Y Khan & P K Jain, “Financial Management”, McGraw Hill, 2018.
5. Ruddar Datt, “Indian Economy”, S.Chand and Company Ltd, 2018.
6. Dwivedi D N, “Macro Economics”, McGraw Hill, 2018.
7. Gregory N Mankiw, “Principles of Micro Economics”, Cengage Learning India, 2020.
8. James C Van Horne, “Financial Management and Policy”, Pearson Education, 20.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Fundamentals of Business Economics		8 hours
1.1	Introduction - Demand and its determinants – Law of demand – Elasticity of demand – Measurement of elasticity and its applications (Numerical problems).	2
1.2	Supply and its determinants – Determination of equilibrium price – Changes in demand and supply and its effects.	2
1.3	Law of diminishing marginal utility - Consumer surplus - producer surplus	2
1.4	Production concepts: Production function - Cobb Douglas function (Numerical problems) - Average product - Marginal product - Law of variable proportions - Law of Returns to Scale.	2
<i>*Activity 1: OPEC decides to reduce its output of oil. Using demand and supply curves bring out the effect of this on the price of oil in the world market.</i> <i>Activity 2: Derive the determination of the equilibrium price of a super luxury and an economy car.</i>		
Module 2: Cost, Revenue and Markets		7 hours
2.1	Concepts of cost - Opportunity cost - Explicit and implicit cost – Private and social cost- Short run cost curves – Fixed, variable, total, average and marginal cost curves - Long run cost curves.	2
2.2	Concepts of revenue – Average and marginal revenue - Shutdown point – Break Even analysis (Numerical problems).	2
2.3	Markets: Perfect Competition, Monopoly, Monopolistic Competition, Oligopoly - Cartel and Collusion (Features and equilibrium of a firm).	2
2.4	Product pricing: Cost plus pricing – Target return pricing – Penetration pricing – Predatory pricing – Going rate pricing – Price skimming – Administered pricing.	1
<i>Activity 1: Determination of equilibrium price and output in oligopoly companies in India.</i> <i>Activity 2: Pricing strategy followed by Apple in regard to their mobiles.</i>		

Module 3: National Income, Inflation and International Trade		7 hours
3.1	Circular flow of income - Multi-sector model - National income concepts – GNP, GDP, NNP, NI, PI, DPI, PCI - Methods of measuring national income – Difficulties (Numerical problems).	2
3.2	Inflation –Types - Causes and effects – Measures to control inflation - Monetary and fiscal policies – Deflation.	2
3.3	International Trade - Balance of payments – Components – Deficit.	2
3.4	Devaluation — Tariff and non-tariff barriers.	1
<i>Activity 1: Compare the present BoP position of India and China.</i> <i>Activity 2: Impact of tariff wars in today's global scenario.</i>		
Module 4: Financial Management		7 hours
4.1	Introduction - Reserve Bank of India – Functions - Credit control techniques: Quantitative and qualitative techniques.	2
4.2	Working capital management - Factors affecting working capital.	1
4.3	Management of cash and marketable securities - Receivables management. Balance Sheet - Profit and Loss Account.	2
4.4	Personal Finance: Budget – Tracking income and expenses - 50-30-20 budgeting rule – Emergency fund – Debit vs Credit Cards – Diversification of Investments – Shares vs Bonds - Power of Compounding – Financial independence – Insurance – Types of Insurance - Digital technology in Finance.	2
<i>Activity 1: Investigate the historical returns offered by different asset classes.</i> <i>Activity 2: Steps needed to circumnavigate financial challenges like student loans, buying a car, purchasing a home vs renting etc.</i>		
Module 5: Business Financing		7 hours
5.1	Introduction - The Stock Market – Functions, Problems faced by the stock market in India – Demat account and trading account – Market indices: Sensex and Nifty.	3
5.2	Derivatives: Forwards, Futures, Options, Swaps - Mutual Funds – Types.	1

5.3	Sources of business financing: Equity capital - Preference capital - Debenture capital - Term loans - Retained earnings.	2
5.4	International Financing - FDI, FII.	1
<i>Activity 1: Research and present the stock performance of a company.</i>		
<i>Activity 2: Investigate the impact of foreign direct investment into India taking the examples of multinational companies.</i>		

* Activities are a desirable part of the course.

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. State the Law of demand.
2. With the help of a figure, elucidate the concept of consumer surplus.
3. Define utility. State the Law of diminishing marginal utility.

Course Outcome 2 (CO 2):

1. Distinguish between explicit and implicit cost.
2. Bring out the relationship between average and marginal revenue.
3. How does a firm under monopoly attain equilibrium?

Course Outcome 3 (CO 3):

1. With the help of a figure, examine the circular flow of income in a multi sector economy.
2. State the government measures to control inflation.
3. What are non-tariff barriers? Give two examples.

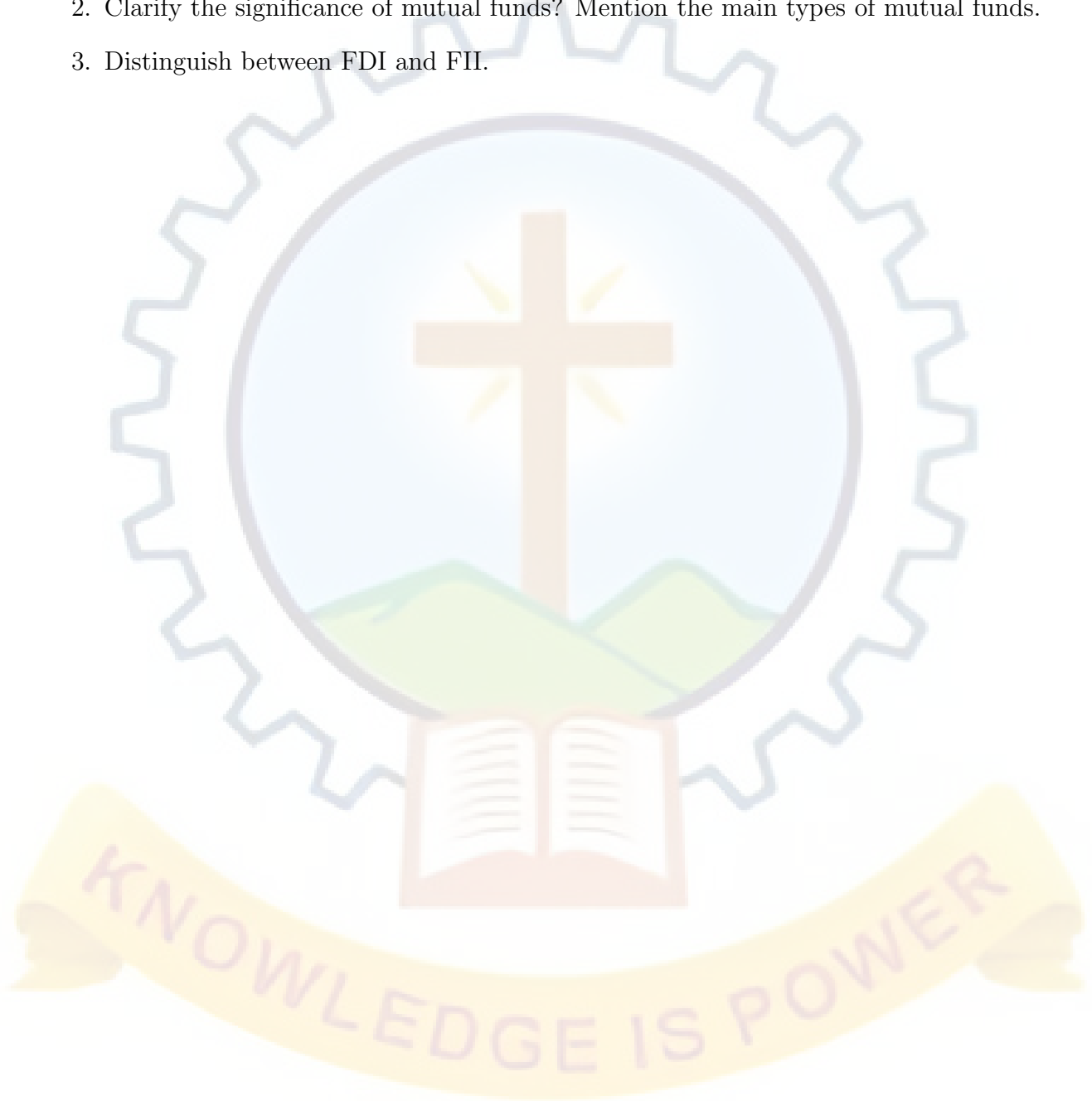
Course Outcome 4 (CO 4):

1. Mention any four functions of the RBI.
2. Elucidate the concept of working capital management. State the main factors influencing it.

3. Clarify the significance of an emergency fund? Mention its advantages.

Course Outcome 5 (CO 5):

1. Elucidate the main problems faced by the stock market in India.
2. Clarify the significance of mutual funds? Mention the main types of mutual funds.
3. Distinguish between FDI and FII.



MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

THIRD SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2025

Course Code: B24HU2T01

***Course Name: BUSINESS ECONOMICS AND FINANCIAL
MANAGEMENT***

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. With the help of a figure, elucidate the concept of consumer surplus.
2. A tea company sold 40000 kg of tea when the price of coffee was Rs.50 per kg. Later they were able to sell 45000 kg when the price of coffee increased to Rs.70 per kg. Calculate the cross elasticity of demand for tea.
3. Distinguish between explicit and implicit cost.
4. List the features of a firm under perfect competition.
5. Define cost plus pricing. Mention its advantage.
6. Write a note on non-tariff barriers. Give two examples.
7. Define Cash Reserve Ratio.
8. Write a note on the significance of receivables management.
9. Distinguish between demat account and trading account.
10. What is FDI? Mention two of its merits.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) State the Law of Demand. Also mention the assumptions and exceptions of the law. (7)
- (b) What is the Cobb- Douglas production function? Mention its feature. If the production function of a firm is $Q = 30L^{1/2}K^{1/2}$, find out the average and marginal product of labour from the function, if 225 units of labour is combined with 196 units of capital. (7)

OR

12. (a) State the Law of diminishing marginal utility. Also mention the assumptions and importance of the law. (7)
- (b) Define Income elasticity. When the income of a consumer was Rs. 50000 per month, the quantity demanded of a good was 100 units. When his income increased to Rs.80000, his demand increased to 110 units. Is the good a normal or an inferior one? Give reason. (7)
13. (a) State the features of a firm under monopolistic competition. With the help of a figure, explain the determination of equilibrium price and output under monopolistic competition. (7)
- (b) A firm sells its product at Rs.400 per unit. To produce a unit, it needs raw materials for Rs. 150, labour for Rs.70 and incurs other variable expenses for Rs. 40. The firm's fixed expenses are Rs.15,00,000. Find the breakeven quantity of the firm. (7)

OR

14. (a) State the features of a firm under perfect competition. With the help of a figure, explain the determination of equilibrium price and output under perfect competition. (7)
- (b) The value of the total sales of a company is Rs. 100000. Its fixed cost is Rs. 20000, while its variable cost is Rs.50000. Calculate
- the P/V ratio
 - breakeven point
 - Margin of safety at this level of sales
 - If it sells each unit for Rs.20, how many units should the company sell to break even?
 - Find the sales required to earn a profit of Rs.20000. (7)
15. (a) With the help of a figure, examine the circular flow of income in a multi sector economy. (7)
- (b) From the following data,
- Gross National Product = Rs 14700 crores
 - GST = Rs 1100 crores

- (c) Undisbursed Profit = Rs 2000 crores
- (d) Corporate Income Tax = Rs 1000 crores
- (e) Depreciation = Rs 1500 crores
- (f) Net Factor Income from abroad = Rs 5200 crores
- (g) Income Tax = Rs 500 crores
- (h) Subsidies = Rs 400 crores
- (i) Social Security Contribution = Rs 300 crores

Calculate

- i. GDP
 - ii. NI
 - iii. PI
- (7)

OR

16. (a) Elucidate the economic problem of inflation. What are its main types? State the government measures to control inflation. (7)
- (b) In an economy, the total expenditure of the people on various goods and services is Rs 2000 crores. The government spending is Rs 500 crores while the total investment is Rs 300 crores. Exports are Rs 200 crores and imports are Rs. 100 crores. The depreciation is Rs 80 crores. Find the value of GNP. (7)
17. (a) Discuss the functions of the RBI. What are the main quantitative techniques used by the RBI? (7)
- (b) State the meaning of balance sheet in accounting. Draw a format of the balance sheet showing the different entries. (7)

OR

18. (a) Write a note on the management of cash and marketable securities. (7)
- (b) State the significance of profit and loss account. Illustrate a format of the profit and loss account. (7)
19. (a) Elaborate the main functions performed by the stock market in an economy. (7)
- (b) Elucidate the various sources of business financing available to companies. (7)

OR

20. (a) Elaborate the meaning of mutual funds. Discuss the different types of mutual funds. (7)
- (b) Distinguish between FDI and FII. (7)

B24CS2L03	DATA STRUCTURES LAB	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	3	2		2024

Preamble

The aim of the Course is to give hands-on experience for Learners on creating and using different Data Structures. Data Structures are used to process data and arrange data in different formats for many applications. The most commonly performed operations on data structures are traversing, searching, inserting, deleting, and a few special operations like merging and sorting.

Prerequisites

Problem Solving and Programming Techniques(A)(B24ES1T01A)

Course Outcomes

After the completion of the course the student will be able to

CO 1	Develop time/space-efficient programs using arrays, queues, circular queues, and priority queues for sorting, searching, arithmetic expression evaluation, polynomial operations, sparse matrices, etc. (Cognitive Knowledge Level: Apply)
CO 2	Implement algorithms and develop programs using linked lists for various user applications. (Cognitive Knowledge Level: Apply)
CO 3	Develop and execute time/space-efficient programs using trees and graphs for various applications. (Cognitive Knowledge Level: Apply)
CO 4	Identify and develop suitable sorting algorithms to sort a given set of records. (Cognitive Knowledge Level: Apply)
CO 5	Experiment with hash tables and apply various collision resolution techniques to solve problems that require fast lookups, insertions, and deletions. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	3	2	1	2	1	1		2	1	1
CO 2	2	3	2	2	2	1	1	2	3	1	2
CO 3	2	3	2	2	3	1	1		3	1	3
CO 4	2	3	3	3	2	1	1	2	2	1	2
CO 5	2	3	2	3	3	1	1	1	3	1	3

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Evaluation	20 marks
- Viva (Self Study)	10 marks
- Output	10 marks
Lab Record	5 marks
Viva voce/Test	15 marks

SYLLABUS

LIST OF EXPERIMENTS

1	Find the sum of two sparse polynomials using arrays.
2	Find the transpose of a sparse matrix and the sum of two sparse matrices. *
3	Convert infix expression to postfix (or prefix) and then evaluate using stack.*
4	Implement Stack, Queue, DEQUEUE, and Circular Queue using arrays.*
5	Implement backward and forward navigation of visited web pages in a web browser (i.e., back and forward buttons) using doubly linked list operations.
6	Implement addition and multiplication of polynomials using singly linked lists.
7	Create a binary tree for a given simple arithmetic expression and find the prefix/postfix equivalent.*
8	Implement a dictionary of word-meaning pairs using binary search trees.*
9	Implement the find and replace feature in a text editor.
10	Implementation of searching Algorithms: Linear Search, Binary Search*
11	Implement Insertion Sort, Quick Sort, Heap Sort and Merge Sort and compare the number of steps involved.*
12	Representation of graphs and computing various parameters (in degree, out degree etc.) - adjacency list, adjacency matrix.

13	The General post office wishes to give preferential treatment to its customers. They have identified the customer categories as Defence personnel, Differently abled, Senior citizen, Ordinary. The customers are to be given preference in the decreasing order - Differently abled, Senior citizen, Defence personnel, Normal person. Generate the possible sequence of completion.
14	Implement a spell checker using a hash table to store a dictionary of words for fast lookup. Implement functions to check if a word is valid and to suggest corrections for misspelled words.*
15	The CSE department is organizing a tech fest with so many exciting events. By participating in an event, you can claim for activity points. Each event i gives you $A[i]$ activity points where A is an array. If you are not allowed to participate in more than k events, calculate the max number of points that you can earn?
16	Merge K sorted lists into a single sorted list using a heap. Use a min-heap to keep track of the smallest element from each list. Repeatedly extract the smallest element and insert the next element from the corresponding list into the heap until all lists are merged.
17	Implement a stack using linked list with the operations: a) Push elements to the queue. b) Pop elements from the queue. c) Display the queue after each operation.
18	Implement a Queue using linked list with the operations: a) Insert an elements to the queue. b) Delete an elements from the queue. c) Display the queue after each operation.
19	Write a program to reverse the content of queue using stack expression.
20	Write a program for addition of polynomials containing two variables using linked list.
21	The details of students(number, name, total-mark) are to be stored in a linked list. Write functions for the following operations: a) Insert b) Delete c) Search d) Sort on the basis of number e) Display the resultant list after every operation

22	Create a Doubly Linked List from a string taking each character from the string. Check if the given string is palindrome in an efficient method.
23	Create a binary tree with the following operations a) Insert a new node b) Inorder traversal. c) Preorder traversal. d) Postorder traversal. e) Delete a node.
24	Write a program to create a binary search tree and find the number of leaf nodes
25	Create a binary search tree with the following operations: a) Insert a new node b) Inorder traversal c) Preorder traversal d) Postorder traversal e) Delete a node
26	Write a program to sort a set of numbers using a binary tree.
27	Represent any given graph and * a) Perform a depth first search b) Perform a breadth first search
28	Create a text file containing the name, height, weight of the students in a class. Perform Quick sort and Merge sort on this data and store the resultant data in two separate files. Also write the time taken by the two sorting methods into the respective files. Eg. Sony Mathew 5.5 60 Arun Sajeev 5.7 58 Rajesh Kumar 6.1 70
29	Write a program to sort a set of numbers using Heap sort and find a particular number from the sorted set using Binary Search.
30	Implement a Hash table using Chaining method. Let the size of hash table be 10 so that the index varies from 0 to 9.
31	Implement a Hash table that uses Linear Probing for collision resolution.

* Mandatory

B24CS2L04	NETWORKING LAB	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	3	2		2024

Preamble

The course enables the learners to get hands-on experience in network programming using file transfer techniques and network monitoring tools. It covers implementation of network protocols and algorithms, configuration of network services and familiarization of network simulators. This helps the learners to develop, implement protocols and evaluate its performance for real world networks.

Prerequisites

Problem Solving and Programming Techniques(A)(B24ES1T01A), Object Oriented Programming(B24CS1T03)

Course Outcomes

After the completion of the course the student will be able to

CO 1	Demonstrate fundamental network programming concepts. (Cognitive Knowledge Level: Apply)
CO 2	Apply file transfer techniques and multi-threading to develop efficient and concurrent network applications. (Cognitive Knowledge Level: Apply)
CO 3	Analyze network traffic and packet flow. (Cognitive Knowledge Level: Apply)
CO 4	Implement and compare various network protocols, including MAC protocols, routing algorithms, and congestion control mechanisms. (Cognitive Knowledge Level: Apply)
CO 5	Analyze network security measures by designing an Intrusion Detection System. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	3	2	2	1	1	1	1	1	1	1
CO 2	3	3	3	2	1	1	1	1	1	1	1
CO 3	3	3	3	3	1	1	1	1	1	1	1
CO 4	3	3	3	3	1	1	1	1	1	1	1
CO 5	3	2	2	2	1	1	1	1	1	1	1

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Evaluation	20 marks
- Viva (Self Study)	10 marks
- Output	10 marks
Lab Record	5 marks
Viva-voce/Test	15 marks

SYLLABUS

LIST OF EXPERIMENTS

1	(a) Create three programs, two of which are clients to a single server. Client1 will send a character to the server process. The server will decrement the letter to the next letter in the alphabet and send the result to Client 2. Client2 prints the letter it receives, and then all the processes terminate. (b) Write a socket program to enable client1 to send a float value to the server. The server process should increase the value of the number it receives by a power of 1.5. The server should print both the value it receives and the value that it sends. Client2 should print the value it receives from the server. (c) Send a C structure that includes data of type character, integer, and float from client1 to the server. The server should change the values so client2 receives a structure with entirely different data. Data should not be converted to any other data type before transmission.*
2	Implement the following using TCP socket: When the server receives a message from a client, it simply converts the message by using following rule: "If a character is a letter or a digit, it will be replaced with the next character in the character set, except that Z will be replaced by A, z by a and 9 by 0. Thus, i becomes j, C becomes D, p becomes q, and so on. Any character other than a letter or a digit will be replaced by a period(.)" and send it back to the client. This sending and receiving message should be repeated until the client and server send a BYEBYE message.*

3	Implement a chat server that allows multiple clients to send and receive messages in real-time using TCP. The server should broadcast messages to all connected clients.
4	Develop a multi-threaded TCP server where each client receives a unique thread to handle requests. Clients should send a number, and the server should return its factorial. Modify the server to handle both TCP and UDP-based requests concurrently.
5	Use Wireshark to capture packets transferred while loading a web page and identify the transport layer protocol used. Analyze the captured packets to extract HTTP request methods and response status codes.*
6	Implement a basic slotted ALOHA protocol where multiple clients try to send packets, and collisions are detected. Extend the program to implement CSMA/CD and simulate collision detection and retransmission.
7	Create three topologies with five nodes and generate FTP traffic for 60 seconds. Generate CBR traffic for a network of 5 nodes and simulate for 150 seconds. Also, create its trace file. Plot TCP throughput using X-graph by varying the number of nodes.
8	Simulate the network for 200 sec and plot the graph of the following: i) TCP throughput vs. simulation time (for every 50secs). ii) TCP packet delays vs. simulation time (for every 50sec). iii) Packet loss vs. simulation time (for every 50sec).*
9	Simulate the Distance Vector Routing with randomly deployed 50 stations. Check for the Count-to-Infinity problem. If it does not exist, then impose the problem and implement any two solutions to resolve the Count-to-Infinity problem in the routing protocol. Simulate 100secs. Measure and plot the network performance in terms of PDR, PLR, routing overhead, and route reconstruction rate for every 10sec interval.*
10	Simulate the link state routing protocol to maintain routing tables as the traffic and topology of the network changes.*

11	a. Simulate the Stop-and-Wait protocol where the sender transmits one packet at a time and waits for an acknowledgment before sending the next packet. Introduce a 5% probability of packet loss and implement retransmission for lost packets. Measure and plot Throughput vs. Simulation Time and Packet Delay vs. Simulation Time for 100 packets. b. Simulate the Go-Back-N protocol with a window size of 4. Implement retransmission of all unacknowledged packets upon detection of a lost packet. Compare the protocol's Throughput and Retransmission Overhead with Stop-and-Wait. c. Simulate the Selective Repeat protocol with a window size of 4, where only lost packets are retransmitted. Introduce a 10% probability of packet loss and measure Packet Delivery Ratio (PDR) vs. Simulation Time. Compare Selective Repeat's performance with Go-Back-N regarding efficiency and delay.*
12	Analyze network performance using Ping and Traceroute commands to measure latency, packet loss, and routing paths. Record and analyze: RTT vs. Packet Loss, RTT vs. Network Delay.*
13	Compare TCP Tahoe and TCP Reno under different traffic conditions using Token Bucket and Leaky Bucket congestion control mechanisms. Simulation Setup Network: 50 nodes, Traffic Levels: Low (5%), Medium (15%), High (25%) Simulation Time: 50 sec (Intervals: 10, 20, 30, 40, 50 sec). Regulate packet transmission based on token availability. Transmit packets at a constant rate, handling bursty traffic. Plot graphs on PDR vs. Simulation Time, Control Overhead vs. Simulation Time, Congestion Rate vs. Simulation Time.*
14	Configure Snort as an Intrusion Detection System (IDS) to monitor network traffic. Capture live packets, detect unauthorized SSH login attempts, and observe alerts. Simulate a DoS attack using excessive ping requests and analyze Snort logs. Perform a port scan using Nmap, detect it and interpret the generated alerts.*

* Mandatory

Reference Books

1. W. Richard Stevens, Bill Fenner, Andy Rudoff, UNIX Network Programming: Volume 1, The Sockets Networking API, 3rd Edition, Pearson, 2015.
2. Lisa Bock, Learn Wireshark: Confidently navigate the Wireshark interface and solve real-world networking problems, Packt Publishing, 2019.
3. Teerawat Issariyakul, Ekram Hossain, Introduction to Network Simulator NS2, 2nd Edition, Springer, 2019.

B24MC2T04	UNIVERSAL HUMAN VALUE AND CONSTITUTIONAL RIGHTS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	0	0	2	P/F	2024

Preamble

This course explores various dimensions of human existence, beginning with self-awareness and an understanding of essential needs such as prosperity, happiness, inner peace, and harmonious relationships. It also introduces the preamble and key features of the Indian Constitution, along with the Directive Principles of State Policy, highlighting their importance in shaping governance and promoting social welfare. By the end of the course, students will be better equipped to act responsibly, address challenges with sustainable solutions, and foster positive human relationships grounded in an understanding of human nature.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Understand the importance of value education for holistic development and to fulfill human aspirations. (Cognitive Knowledge Level: Understand)
CO 2	Develop more awareness of themselves and their surroundings(family, society, nature)to build harmonious and respectful relationships.(Cognitive Knowledge Level: Apply)
CO 3	Understand and appreciate the preamble and other features in the Indian Constitution to promote responsible citizenship. (Cognitive Knowledge Level: Understand)
CO 4	Understand the fundamental rights and duties enshrined in the Indian Constitution and the Directive Principles of State Policy and their role in shaping governance and social welfare. (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1						2	3	1	1	1	2
CO 2						2	3	2	1	1	2
CO 3						2	3		2		2
CO 4						2	3		2		2

Assessment Pattern

Bloom's Category	Continuous Assessment	End Semester Examination (% Marks)
	Test 1 (%Marks)	
Remember	30	30
Understand	60	60
Apply	10	10
Analyse		
Evaluate		
Create		

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	50	50	2 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (One test, conducted for 50 marks and reduced to 25)	25 marks
Assignment/Quiz/Course Project/Seminar	10 marks
Self Study	5 marks

End Semester Examination Pattern

There will be two parts, Part A and Part B. Part A contains 4 questions carrying 3 marks from each module. Part B contains 2 questions from each module out of which one is to be answered. In Part B, each question of first two modules carries 9 marks and each question of last two modules carries 10 marks.

SYLLABUS

MODULE 1 (6 hours)

Introduction to Values

The Need of Value Education-Guidelines for Value Education, Self-exploration as the Process for Value Education- Two parts, Important implications of Self Exploration, Continuous Happiness and Prosperity - A Look at Basic Human Aspirations-Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education)- Method to fulfill the Basic Human Aspirations.

MODULE 2 (5 hours)

Harmony in the Human Being, Family and Society

Understanding Human being as the Co-existence of the Self and the Body- Distinguishing between the Needs of the Self and the Body- Harmony of the Self with the Body- Harmony in the Family – the Basic Unit of Human Interaction- Understanding Harmony in the Society.

MODULE 3 (5 hours)

Introduction to Constitution of India

Definition and Historical Background of the Constitution-Salient Features of the Constitution- Preamble of the Constitution- Union and Its Territory- Meaning and Types of Citizenship- Termination of Citizenship.

MODULE 4 (8 hours)

State Policies and Fundamental Rights

Definition of the State- Fundamental Rights- General Nature and Classification- Right to Equality and Right to Freedom- Right Against Exploitation- Right to Freedom of Religion- Cultural and Educational Rights- Right to Constitutional Remedies- Protection Against Conviction for Offences- Right to Information (RTI) and Its Applications- Directive Principles of State Policy- Classification of Directives- Fundamental Duties.

Text Books

1. A Foundation Course in Human Values and Professional Ethics, R R Gaur, R Asthana, G P Bagaria, 3rd Revised Edition, Excel Books, New Delhi, 2023.
2. The Teacher's Manual for A Foundation Course in Human Values and Professional Ethics, R R Gaur, R Asthana, G P Bagaria, 3rd Revised Edition, Excel Books, New

Delhi, 2023.

3. D D Basu, Introduction to the constitution of India, Lexis Nexis, New Delhi, 26/e, 2022.
4. P M Bhakshi, The constitution of India, Universal Law, 19/e, 2023.

Reference Books

1. M Govindarajan, S Natarajan and V S Senthil Kumar, Engineering Ethics, PHI Learning Private Ltd, New Delhi, 2012.
2. R S Naagarazan, A text book on professional ethics and human values, New age international (P) limited, New Delhi, 2006.
3. Ministry of Law and Justice, The constitution of India, Govt of India, New Delhi, 2019.
4. J N Pandey, The constitutional Law of India, Central Law Agency, Allahabad, 51e, 2019.
5. M V Pylee, Indias Constitution, S Chand and Company, New Delhi, 16e, 2016.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		24 Hours
Module 1: Introduction to Values		6 hours
1.1	The Need of Value Education-Guidelines for Value Education	1
1.2	Self-exploration as the Process for Value Education-Two Parts	1
1.3	Important implications of Self Exploration	1
1.4	Continuous Happiness and Prosperity -A Look at Basic Human Aspirations	1
1.5	Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education)	1
1.6	Method to fulfill the Basic Human Aspirations	1
Module 2: Harmony in the Human Being, Family and Society		5 hours

2.1	Understanding Human being as the Co-existence of the Self and the Body	1
2.2	Distinguishing between the Needs of the Self and the Body	1
2.3	Harmony of the Self with the Body	1
2.4	Harmony in the Family – the Basic Unit of Human Interaction	1
2.5	Understanding Harmony in the Society	1
Module 3: Introduction to Constitution of India		5 hours
3.1	Definition and Historical Background of the Constitution	1
3.2	Salient Features of the Constitution	1
3.3	Preamble of the Constitution-Union and Its Territory	1
3.4	Meaning and Types of Citizenship	1
3.5	Termination of Citizenship	1
Module 4: State Policies and Fundamental Rights		8 hours
4.1	Definition of the State- Fundamental Rights- General Nature and Classification	1
4.2	Right to Equality and Right to Freedom-Right Against Exploitation- Right to Freedom of Religion	1
4.3	Cultural and Educational Rights- Right to Constitutional Remedies	1
4.4	Protection Against Conviction for Offences	1
4.5	Right to Information (RTI) and Its Applications	1
4.6	Directive Principles of State Policy	1
4.7	Classification of Directives	1
4.8	Fundamental Duties	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the basic guidelines for value education. What is the need for these guidelines.
2. Explain the process of self-exploration. What is the expected result of self-exploration?
3. What are the basic human aspirations and what are the requirements to fulfill them? Support your answer with two examples.

Course Outcome 2 (CO 2):

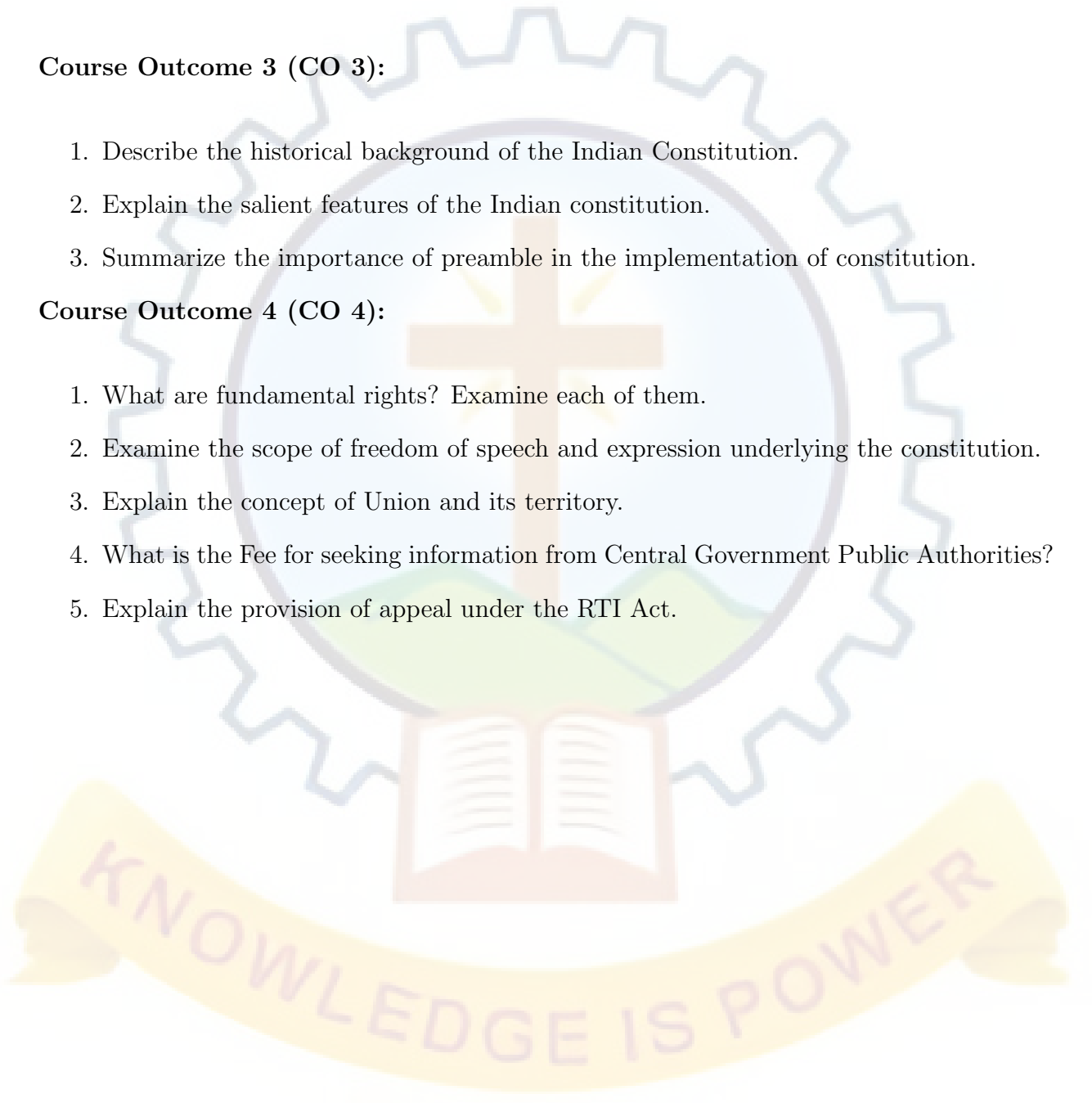
1. Distinguish between 'animal consciousness' and 'human consciousness'.
2. 'Relationship is – between one Self (I1) and another Self (I2)'. Examine this statement.
3. What is the building block for harmony in the society? Explain with examples.

Course Outcome 3 (CO 3):

1. Describe the historical background of the Indian Constitution.
2. Explain the salient features of the Indian constitution.
3. Summarize the importance of preamble in the implementation of constitution.

Course Outcome 4 (CO 4):

1. What are fundamental rights? Examine each of them.
2. Examine the scope of freedom of speech and expression underlying the constitution.
3. Explain the concept of Union and its territory.
4. What is the Fee for seeking information from Central Government Public Authorities?
5. Explain the provision of appeal under the RTI Act.



KNOWLEDGE IS POWER

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

THIRD SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2025

Course Code: B24MC2T04

Course Name: Universal Human Values and Constitutional Rights

Max. Marks: 50

Duration: 2 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Identify the solution which helps human being to transform from animal Consciousness to human consciousness.
2. What is the building block for harmony in the society?
3. Define and explain the term Constitution.
4. "The freedom of speech and expression is subject to reasonable restrictions". Explain the statement.

PART B

Answer any one question from each module.

5. Explain the basic guidelines for value education. What is the need for these guidelines.
(9)

OR

6. Choose any five things that you consider as human values. Write all the basic guidelines, and check if they satisfy the basic guidelines.
(9)

7. Distinguish between 'animal consciousness and 'human consciousness'. (9)

OR

8. "Relationship is – between one Self(I1) and another Self(I2)". Examine this statement. (9)

9. Summarize the various methods of acquiring Indian citizenship. (10)

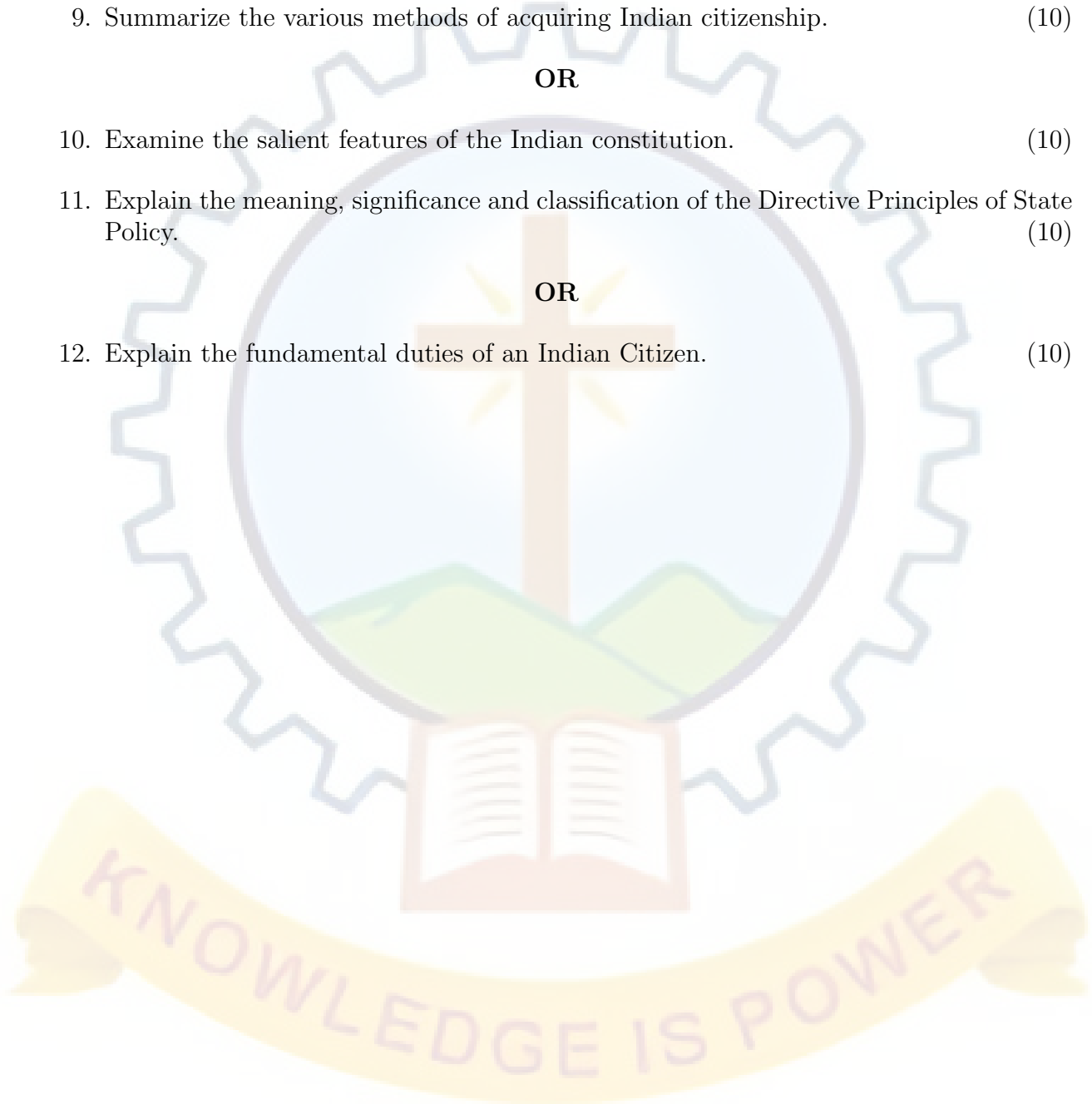
OR

10. Examine the salient features of the Indian constitution. (10)

11. Explain the meaning, significance and classification of the Directive Principles of State Policy. (10)

OR

12. Explain the fundamental duties of an Indian Citizen. (10)



B24MC2T05	ENERGY CONSERVATION AND ENVIRONMENTAL SUSTAINABILITY	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	0	0	2	P/F	2024

Preamble

This course aims to equip students with fundamental knowledge of energy resources, the need for energy conservation, and the importance of environmental sustainability. It emphasizes the role of engineers in adopting renewable energy technologies, reducing environmental impact, and promoting sustainable development for a greener and more resilient future.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Classify various energy resources and describe the importance of energy conservation (Cognitive Knowledge Level: Understand)
CO 2	Explain the principles of renewable energy systems and their applications (Cognitive Knowledge Level: Understand)
CO 3	Recognize major environmental impacts due to energy consumption and explain basic pollution control measures. (Cognitive Knowledge Level: Understand)
CO 4	Describe sustainability concepts and apply simple strategies for environmental protection and green practices in day-to-day engineering tasks (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	1				2	2				
CO 2	3	1	1			1	2				
CO 3	2	2				2	3	1			
CO 4	2	1	2			2	3	1	1	1	

Assessment Pattern

Bloom's Category	Continuous Assessment	End Semester Examination (% Marks)
	Test 1 (%Marks)	
Remember	30	30
Understand	50	50
Apply	20	20
Analyse		
Evaluate		
Create		

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	50	50	2 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (1)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contains 4 questions carrying 3 marks each. Part B contains 2 questions from each module out of which one has to be answered. In part B, each question of the first two modules carries 9 marks and each question of the last two modules carries 10 marks.

SYLLABUS

MODULE 1 (6 hours)

Energy Resources and Conservation

Types of Energy Resources: Renewable and Non-renewable (with examples), Global and Indian Energy Scenarios, Importance of Energy Conservation, Energy Policy and Planning.

Energy Auditing and Efficiency Improvement Techniques - Case studies, Energy-efficient Buildings and Smart Cities.

MODULE 2 (6 hours)

Renewable Energy Technologies

Solar Energy, Wind Energy, Other Renewable Sources: Biomass and bioenergy systems, small hydropower, ocean thermal, wave, and tidal energy, Geothermal energy.

Energy Storage and Smart Grid Integration: Battery technologies - Role in renewable energy conservation and stability - Decentralized generation and net metering, Advanced and Emerging Technologies: Green hydrogen - Floating solar farms - Offshore wind.

MODULE 3 (6 hours)

Environmental Impact and Pollution Control

Pollution Types and Sources: Air pollution: industrial emissions, vehicular sources - Water pollution: domestic, industrial, agricultural waste - Soil pollution: hazardous waste, agrochemicals, Pollution Control Methods: Physical, chemical, biological techniques - Air & water treatment technologies.

Climate Change and Global Warming: Greenhouse gases and carbon footprint - International agreements, Waste Management Strategies - 3Rs (Reduce, Reuse, Recycle) - waste-to-energy, Environmental Regulations in India: Environmental Protection Act, Air & Water Acts - Hazardous Waste Management Rules, Carbon Neutrality and Zero-emission Policies.

MODULE 4 (6 hours)

Sustainability and Green Practices

Principles of Sustainable Development: Intergenerational equity, resource efficiency - Link with UN Sustainable Development Goals (SDGs), Green Buildings and Infrastructure, Green Certification Systems, Carbon Credits.

Carbon Pricing and Energy Subsidies: Internal carbon pricing by organizations - Government schemes, Life Cycle Assessment (LCA): Phases of LCA: Goal definition, inventory, impact assessment - Smart Sustainable Cities and Resilient Infrastructure: Urban planning for sustainability.

Text Books and References

1. Charles M. Gottschalk, Industrial Energy Conservation, John Wiley & Sons, 1996.
2. G.D. Rai, Non-Conventional Energy Sources, Khanna Publishers
3. R.R. Rao, Environmental Science and Engineering, PHI
4. Craig B. Smith, Energy Management Principles, Pergamon Press.
5. Paul O'Callaghan, Energy Management, McGraw Hill Book Co.
6. Wayne C. Turner, Energy Management Hand Book, The Fairmount Press, Inc., 1997.
7. Allen, D. T. and Shonnard, D. R., Sustainability Engineering: Concepts, Design and Case Studies, Prentice Hall.
8. Bradley. A.S; Adebayo, A.O., Maria, P. Engineering applications in sustainable design and development, Cengage learning.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		24 Hours
Module 1: Energy Resources and Conservation		6 hours
1.1	Types of Energy Resources (Renewable and Non-renewable), examples. Global and Indian Energy Scenario: Consumption and production trends, sector-wise demand	1
1.2	Energy Conservation: Residential, Industrial, Transport sectors – behavioural & technological interventions	1
1.3	Energy Policy and Planning: National Energy Policy, Energy Conservation Act, BEE initiatives	1
1.4	Energy Auditing: Preliminary & Detailed audits, Performance Indicators, Case Studies	2

1.5	Energy-efficient Buildings & Smart Cities: Passive design, daylighting, automation	1
Module 2: Renewable Energy Technologies		6 hours
2.1	Solar Energy: PV systems, solar thermal, rooftop/grid-tied applications	1
2.2	Wind Energy: Onshore/offshore systems, hybrid solar-wind systems	1
2.3	Other Renewables: Biomass, small hydro, ocean, tidal, geothermal energy	1
2.4	Energy Storage: Battery types, conservation role, stability, smart grid integration	1
2.5	Smart Grids and Net Metering: Decentralized generation	1
2.6	Advanced Technologies: Green hydrogen, floating solar farms, offshore wind	1
Module 3: Environmental Impact and Pollution Control		6 hours
3.1	Pollution Types: Air, water, and soil pollution – sources and effects	1
3.2	Pollution Control: Physical, chemical, and biological treatment methods	1
3.3	Climate Change: GHGs, carbon footprint, international agreements (Kyoto, Paris)	1
3.4	Waste Management: Solid, liquid, biomedical, hazardous – 3Rs, waste-to-energy, Environmental regulations	1
3.5	Carbon Neutrality: National missions, zero-emission policies, corporate initiatives	1
3.6	Circular Economy: Waste elimination	1
Module 4: Sustainability and Green Practices		6 hours
4.1	Sustainable Development: Principles, SDGs, resource efficiency	1
4.2	Green Buildings: Concepts, features, materials, passive design, renewables integration, green certifications	1
4.3	Carbon Credits: Earning, trading, CDM, voluntary carbon markets	1
4.4	Carbon Pricing and Subsidies: Internal pricing, UJALA, PM-KUSUM, FAME	1
4.5	Life Cycle Assessment (LCA): Phases, case studies	1

4.6	Smart Cities and Resilience: Urban planning	1
<i>Students shall present a seminar based on case studies of Life Cycle Assessment (LCA) conducted on a product of their choice.</i>		

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. List the major renewable and non-renewable energy sources with suitable examples.
2. Explain the significance of energy conservation in the industrial and domestic sectors.
3. Describe the importance of energy conservation in the context of the global and Indian energy scenario.

Course Outcome 2 (CO 2):

1. Describe the working principle of solar photovoltaic and solar thermal systems.
2. Compare wind energy and small hydropower systems based on availability, reliability, and applications.
3. Explain the role of energy storage and smart grid integration in ensuring renewable energy reliability.

Course Outcome 3 (CO 3):

1. Identify major sources of air and water pollution in urban areas.
2. Explain the role of battery storage and smart grid integration in enhancing the efficiency of renewable energy systems.
3. Apply the concept of 3Rs to develop a basic household or institutional waste management plan.

Course Outcome 4 (CO 4):

1. Describe the concept of sustainable development and its connection with UN Sustainable Development Goals (SDGs).
2. Explain the basic features of green buildings and the benefits of green certification.
3. Describe simple green practices that can be adopted by engineers in daily professional work to promote environmental sustainability.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM

THIRD SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2025

Course Code: B24MC2T05

*Course Name: ENERGY CONSERVATION AND ENVIRONMENTAL
SUSTAINABILITY*

Max. Marks: 50

Duration: 2 hours

PART A

Answer all questions. Each question carries 3 marks.

1. List any three renewable energy sources with one example each.
2. Explain the concept of net metering in decentralized energy generation.
3. What are the major sources of air pollution in urban areas?
4. Describe any two strategies for promoting sustainability in everyday engineering practices.

PART B

Answer any one question from each module.

5. (a) Classify energy resources with examples. (4)
(b) Explain the significance of energy conservation in the Indian context. (5)

OR

6. (a) Describe energy auditing and mention any two efficiency improvement techniques. (5)

- (b) What is the role of energy-efficient buildings in smart city development? (4)
7. (a) Explain the working principle of wind turbines with a neat diagram. (5)
- (b) Describe any two advanced renewable energy technologies. (4)

OR

8. (a) Discuss the types and role of energy storage systems in renewable energy. (5)
- (b) Explain the concept of smart grid integration. (4)
9. (a) What are the major environmental impacts of energy consumption? (4)
- (b) Explain physical, chemical, and biological methods of water pollution control. (4)

OR

10. (a) Describe the working of waste-to-energy technologies. (5)
- (b) List and briefly explain any two environmental regulations in India. (5)
11. (a) Define sustainable development and explain its principles. (4)
- (b) What are carbon credits and how do they promote sustainability (6)

OR

12. (a) Explain the key phases of Life Cycle Assessment (LCA). (5)
- (b) What is the importance of green buildings and certification systems in achieving sustainable urban infrastructure? (5)

B24CSM31	PYTHON PROGRAMMING (Minor)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course provides a comprehensive introduction to Python, covering fundamental programming concepts, selection structures, functions, loops, data structures, recursion, and file handling. By the end of the course, learners will have gained essential programming skills, enabling them to develop efficient algorithms and work with real-world data effectively.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Demonstrate an understanding of fundamental programming concepts, selection structures, and basic graphics programming in Python, applying engineering fundamentals and natural science principles (e.g., data representation, algorithmic thinking) to solve computational problems. (Cognitive knowledge level: Apply)
CO 2	Implement and analyze functions, loops, and string manipulations to solve computational problems efficiently, using mathematical and statistical concepts for data processing and analysis. (Cognitive knowledge level: Apply)
CO 3	Utilize data structures such as lists, tuples, dictionaries, and sets to store, retrieve, and manipulate data effectively, while considering resource efficiency and reproducibility in program execution. (Cognitive knowledge level: Apply)
CO 4	Develop modular programs using functions, recursion, and abstraction techniques to enhance code reusability and problem-solving skills, demonstrating ethical and inclusive practices in software development. (Cognitive knowledge level: Apply)
CO 5	Handle files and exceptions in Python, including text and binary file operations, to manage data and ensure robust program execution, while engaging with current research literature and evaluating societal implications of data-driven solutions (e.g., privacy, data integrity). (Cognitive knowledge level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	1	2	3	0	0	0	0	0	1
CO 2	2	3	1	3	3	0	0	0	0	0	0
CO 3	2	3	2	3	3	1	0	0	0	2	0
CO 4	2	3	2	3	3	1	3	3	0	0	0
CO 5	1	3	2	3	3	3	0	0	3	0	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember			
Understand	20	20	20
Apply	40	40	40
Analyse	40	40	40
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction to Python and Selection Structures

Programming Languages and Operating Systems, The History of Python, Getting Started with Python, Programming Style and Documentation.

Selections-Boolean Types, Values, and Expressions, Generating Random Numbers, if Statements, if-else, Nested if, Conditional Expressions, Logical Operators and Operator Precedence.

Case Studies: BMI Calculation, Computing Taxes, Leap Year, Lottery.

MODULE 2 (9 hours)

Functions, Strings, and Loops

Mathematical Functions, Strings, and Objects, Common Python Functions, Strings and Characters, String Methods, Formatting Numbers and Strings, Introduction to Objects and Methods.

Case Study: Revising the Lottery Program.

Loops -The while Loop and for Loop, Nested Loops, Loop Design Strategies, Controlling a Loop with User Confirmation, break and continue Statements.

Case Study: Checking Palindromes, Displaying Prime Numbers.

MODULE 3 (10 hours)

Lists, Tuples, Set and Dictionaries

Lists - List Basics, Copying, Searching, and Sorting Lists, Passing Lists to Functions, Returning Lists from Functions.

Case Study: Analyzing Numbers, Counting the Occurrences of Each Letter.

Tuples, Sets, Comparing the Performance of Sets and Lists, Dictionaries.

Case Study: Counting Keywords. Dictionaries and Word Occurrences.

MODULE 4 (9 hours)

Functions and Recursion

Functions- Defining and Calling Functions, Functions with/without Return Values, Positional and Keyword Arguments, Passing Arguments by Reference Values, Modularizing Code, The Scope of Variables. Function Abstraction and Stepwise Refinement.

Case Study: Generating ASCII Characters, Converting Hexadecimal to Decimal. Recursion.Problem Solving Using Recursion, Recursive Helper Functions.Recursion vs. Iteration, Tail Recursion.

Case Study: Computing Factorials, Fibonacci Numbers.

MODULE 5 (10 hours)

Files and Exception Handling

Files and Exception Handling -Text Input and Output, File Dialogs.Modules. Case Study: Counting Each Letter in a File, Retrieving Data from the Web

Exception Handling Basics, Raising Exceptions, Processing Exceptions Using Exception Objects, Defining Custom Exception Classes, Binary I/O Using Pickling.

Case Study: Address Book.

Text Books

1. Introduction to Python Programming and Data Structures by Y. Daniel Liang.

Reference Books

1. Python Programming: An Introduction to Computer Science by John Zelle.
2. Python Crash Course by Eric Matthes.
3. Think Python: How to Think Like a Computer Scientist by Allen B. Downey.
4. Data Structures and Algorithms in Python by Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser.
5. Automate the Boring Stuff with Python by Al Sweigart.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction to Python and Selection Structures		7 hours
1.1	Programming Languages and Operating Systems, The History of Python	1
1.2	Getting Started with Python, Programming Style and Documentation.	1
1.3	Selections-Boolean Types, Values, and Expressions.	1
1.4	Generating Random Numbers	1
1.5	if Statements, if-else, Nested if	1
1.6	Conditional Expressions	1
1.7	Logical Operators and Operator Precedence.	1
Module 2: Functions, Strings and Loops		9 hours
2.1	Mathematical Functions, Strings, and Objects	1
2.2	Common Python Functions	1
2.3	Strings and Characters, String Methods, Formatting Numbers and Strings,	1
2.4	Introduction to Objects and Methods.Case Study: Revising the Lottery Program.	1
2.5	Loops -The while Loop and for Loop	1
2.6	Nested Loops, Loop Design Strategies	1
2.7	Controlling a Loop with User Confirmation	1
2.8	break and continue Statements	1
Case Study: Checking Palindromes, Displaying Prime Numbers (1 hour)		
Module 3: Lists, Tuples, Set and Dictionaries		10 hours
3.1	List Basics	1
3.2	Copying, Searching, and Sorting Lists	1
3.3	Passing Lists to Functions	1
3.4	Returning Lists from Functions	1

Case Study: Analyzing Numbers (1 hour)		
3.5	Counting the Occurrences of Each Letter	1
3.6	Tuples, Sets	1
3.7	Comparing the Performance of Sets and Lists	1
3.8	Dictionaries	1
Case Study: Counting Keywords, Word Occurrences (1 hour)		
Module 4: Functions and Recursion		9 hours
4.1	Defining and Calling Functions	1
4.2	Functions with/without Return Values	1
4.3	Positional and Keyword Arguments, Passing Arguments by Reference Values	1
4.4	Modularizing Code, The Scope of Variables	1
4.5	Function Abstraction and Stepwise Refinement	1
Case Study: Generating ASCII Characters, Converting Hexadecimal to Decimal (1 hour)		
4.6	Recursion, Problem Solving Using Recursion, Recursive Helper Functions	1
4.7	Recursion vs. Iteration, Tail Recursion	1
Case Study: Computing Factorials, Fibonacci Numbers (1 hour)		
Module 5: Files and Exception Handling		10 hours
5.1	Text Input and Output	1
5.2	File Dialogs, Modules	1
Case Study: Counting Each Letter in a File(1 hour)		
5.3	Retrieving Data from the Web	1
5.4	Exception Handling Basics	1
5.5	Raising Exceptions	1
5.6	Processing Exceptions Using Exception Objects	1
5.7	Defining Custom Exception Classes	1
5.8	Binary I/O Using Pickling	1
Case Study: Address Book(1 hour)		

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the impact of programming languages on computational problem-solving and how Python simplifies this process.
2. Discuss the significance of structured programming and how Python enforces structured coding practices.

Course Outcome 2 (CO 2):

1. Explain the difference between iterative and recursive approaches in problem-solving, with examples.
2. Discuss how string manipulation techniques are used in data processing applications such as text analytics.

Course Outcome 3 (CO 3):

1. Explain how Python lists can be used to implement stack and queue data structures.
2. Discuss the applications of dictionaries in real-world scenarios such as database indexing and caching.

Course Outcome 4 (CO 4):

1. Discuss the role of higher-order functions in Python and explain their significance in functional programming.
2. Explain the advantages and challenges of recursive algorithms compared to iterative approaches.

Course Outcome 5 (CO 5):

1. Discuss how file handling in Python is used for data persistence in real-world applications.
2. Explain the role of exception handling in debugging and improving the reliability of Python programs.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

THIRD SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2025

Course Code: B24CSM31

Course Name: PYTHON PROGRAMMING (MINOR)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. What are the different types of programming errors in Python?
2. Explain the significance of the if-else statement in decision-making
3. What is the difference between while and for loops in Python?
4. Explain any two string methods in Python with examples.
5. How does a Python dictionary differ from a list?
6. Explain the significance of the set data structure in Python.
7. What is the difference between positional and keyword arguments in Python functions?
8. Explain the concept of tail recursion with an example.
9. What is the difference between text I/O and binary I/O in Python?
10. Explain how exception handling improves program robustness in Python.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Discuss the evolution of Python as a programming language and compare it with other popular languages. (7)
- (b) Explain the role of selection structures (if, if-else, nested if) in decision-making, with relevant examples like tax computation. (7)

OR

12. (a) Describe different types of programming errors in Python. How can they be identified and resolved? (7)
- (b) Explain the concept of Boolean expressions and logical operators in Python with practical use cases. (7)
13. (a) Discuss the importance of functions in Python programming. How do functions improve code modularity and reusability? (7)
- (b) Explain different loop control mechanisms in Python (break, continue, while, for) with case studies. (7)

OR

14. (a) Compare and contrast string manipulation techniques in Python with examples. (7)
- (b) Discuss how mathematical functions and string formatting techniques are applied in real-world scenarios. (7)
15. (a) Compare and contrast lists, tuples, and dictionaries in Python. Explain their advantages and best-use cases. (7)
- (b) Discuss various list operations (insertion, deletion, sorting, searching) and their applications in real-world scenarios. (7)

OR

16. (a) Explain how dictionaries in Python are used for mapping data efficiently. Provide examples of word count applications. (7)
- (b) Discuss the performance differences between sets and lists in Python and explain when to use each data structure. (7)
17. (a) Discuss function abstraction and stepwise refinement in Python, explaining how it improves software development. (7)
- (b) Explain recursion with case studies on factorial computation, Fibonacci series, and the Tower of Hanoi. (7)

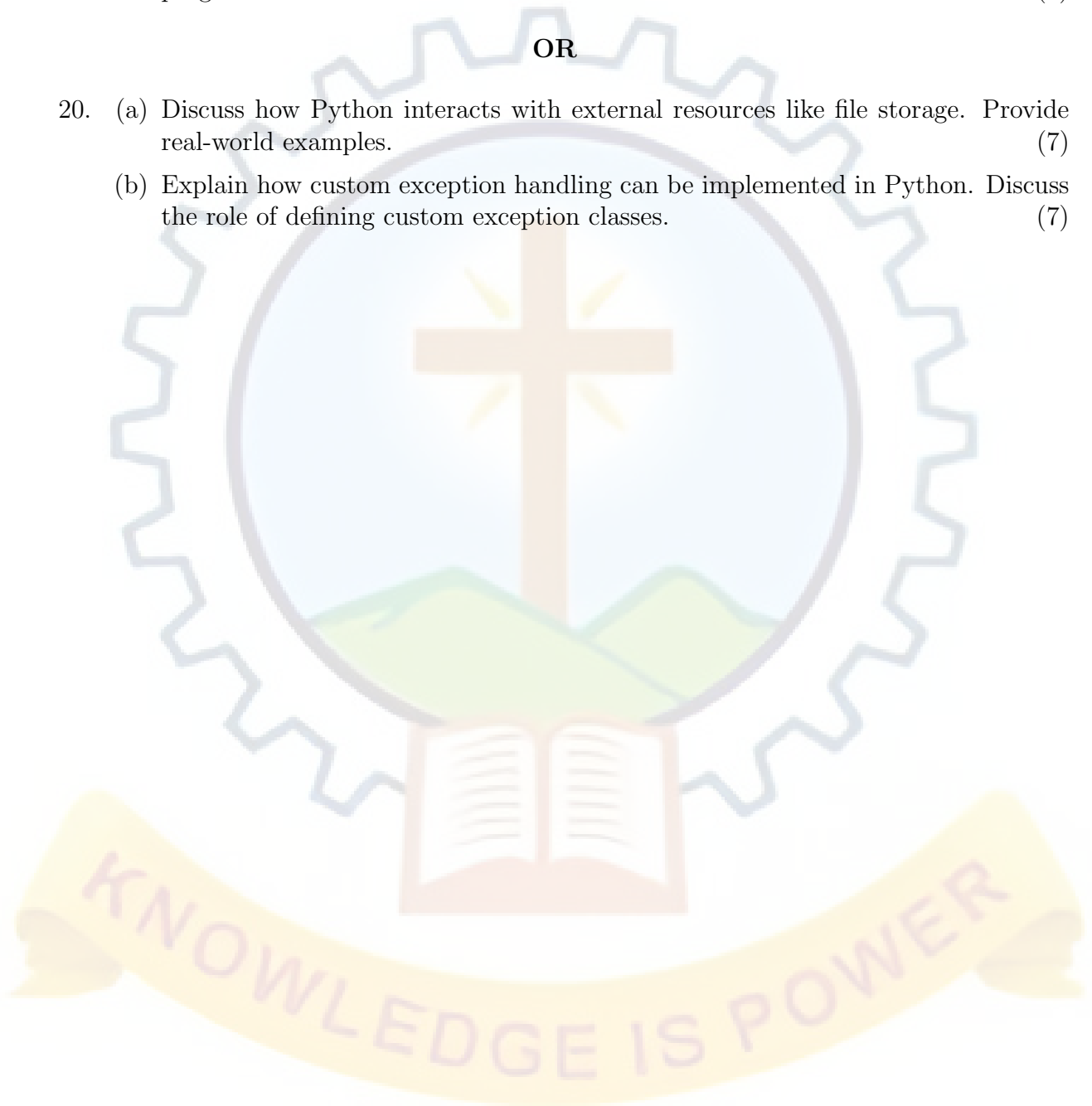
OR

18. (a) Compare iteration and recursion in Python, discussing their advantages and disadvantages. (7)

- (b) Explain the scope of variables in Python functions. How does variable scope impact program execution and debugging? (7)
19. (a) Discuss the importance of file handling in Python. Explain how text and binary files are processed with examples. (7)
- (b) Explain the concept of exception handling in Python and how it enhances program robustness. Provide case studies. (7)

OR

20. (a) Discuss how Python interacts with external resources like file storage. Provide real-world examples. (7)
- (b) Explain how custom exception handling can be implemented in Python. Discuss the role of defining custom exception classes. (7)



B24CSM32	MATHEMATICS FOR MACHINE LEARNING (Minor)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		

Preamble

This course provides learners the mathematical foundation necessary for understanding and developing machine learning models. The course covers Linear Algebra, Vector Calculus, Probability and Distributions, Optimization and Machine Learning problems. Since machine learning relies heavily on concepts from several branches of mathematics, this course equips students with essential skills to grasp the underlying principles of various algorithms and aid in the creation of new Machine Learning solutions, understand & debug existing ones, and learn about the inherent assumptions & limitations of the current methodologies.

Prerequisites

A sound background in higher secondary school Mathematics.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Apply the concepts of linear equations, matrices, vector spaces, linear independence, basis, and rank to solve problems, and use linear mappings, matrix representation, and basis changes in computational applications. (Cognitive Knowledge Level: Apply)
CO 2	Apply concepts from analytic geometry, matrix decompositions, orthogonality, eigenvalues, eigenvectors, and singular value decomposition (SVD) to solve problems related to matrix approximations, dimensionality reduction, and data transformations.(Cognitive Knowledge Level: Apply)
CO 3	Perform calculus operations on functions of several variables and matrices, including partial derivatives and gradients (Cognitive Knowledge Level: Apply)
CO 4	Utilize the concepts, rules and results about probability, random variables, additive and multiplicative rules, conditional probability, probability distributions and Bayes theorem to find solutions of computational problems (Cognitive Knowledge Level: Apply)
CO 5	Train Machine Learning Models using unconstrained and constrained optimization methods (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	2	2	2							1
CO 2	3	2	2								1
CO 3	3	2	2	2							1
CO 4	3	2	2	2							1
CO 5	3	2	2	2							1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (8 hours)

Linear Algebra

Systems of Linear Equations – Matrices, Solving Systems of Linear Equations. Vector Spaces –Vector Spaces, Linear Independence, Basis, and Rank. Linear Mappings – Matrix Representation of Linear Mappings, Basis Change, Image and Kernel.

MODULE 2 (11 hours)

Analytic Geometry, Matrix Decompositions

Norms, Inner Products, Lengths and Distances, Angles and Orthogonality, Orthonormal Basis, Orthogonal Complement, Orthogonal Projections – Projection into One-Dimensional Subspaces, Projection onto General Subspaces, Gram-Schmidt Orthogonalization. Determinant and Trace, Eigen values and Eigen vectors, Cholesky Decomposition, Eigen decomposition and Diagonalization, Singular Value Decomposition, Matrix Approximation.

MODULE 3 (9 hours)

Vector Calculus

Differentiation of Univariate Functions - Partial Differentiation and Gradients, Gradients of Vector Valued Functions, Gradients of Matrices, Useful identities for Computing Gradients. Back propagation and Automatic Differentiation – Gradients in Deep Network, Automatic Differentiation. Higher Order Derivatives- Linearization and Multivariate Taylor Series.

MODULE 4 (10 hours)

Probability and Distributions

Construction of a Probability Space - Discrete and Continuous Probabilities, Sum Rule, Product Rule, and Bayes' Theorem. Summary Statistics and Independence – Gaussian Distribution - Conjugacy and the Exponential Family - Change of Variables/Inverse Transform.

MODULE 5 (7 hours)

Optimization

Optimization Using Gradient Descent - Gradient Descent With Momentum, Stochastic Gradient Descent. Constrained Optimization and Lagrange Multipliers - Convex Optimization - Linear Programming - Quadratic Programming.

Text Books

1. Mathematics for Machine Learning by Marc Peter Deisenroth, A. Aldo Faisal, and Cheng Soon Ong published by Cambridge University Press.

Reference Books

1. Linear Algebra and Its Applications, 4th Edition by Gilbert Strang
2. Introduction to Applied Linear Algebra by Stephen Boyd and Lieven Vandenberghe, 2018 published by Cambridge University Press.
3. Convex Optimization by Stephen Boyd and Lieven Vandenberghe, 2004 published by Cambridge University Press.
4. Pattern Recognition and Machine Learning by Christopher M Bishop, 2006, published by Springer.
5. Learning with Kernels – Support Vector Machines, Regularization, Optimization, and Beyond by Bernhard Scholkopf and Smola, Alexander J Smola, 2002, published by MIT Press.
6. Information Theory, Inference, and Learning Algorithms by David J. C MacKay, 2003. published by Cambridge University Press.
7. Machine Learning: A Probabilistic Perspective by Kevin P Murphy, 2012 published by MIT Press.
8. The Nature of Statistical Learning Theory by Vladimir N Vapnik, 2000, published by Springer.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Linear Algebra		8 hours
1.1	Matrices, Solving Systems of Linear Equations	1
1.2	Vector Spaces	1
1.3	Linear Independence	1
1.4	Basis and Rank (Lecture – 1)	1
1.5	Basis and Rank (Lecture – 2)	1
1.6	Linear Mappings	1
1.7	Matrix Representation of Linear Mappings	1
1.8	Basis Change, Image and Kernel	1
Module 2: Analytic Geometry, Matrix Decompositions		11 hours
2.1	Norms, Inner Products,	1
2.2	Lengths and Distances, Angles and Orthogonality	1
2.3	Orthonormal Basis, Orthogonal Complement	1
2.4	Orthogonal Projections – Projection into One-Dimensional Subspaces,	1
2.5	Projection onto General Subspaces,	1
2.6	Gram-Schmidt Orthogonalization	1
2.7	Determinant and Trace Eigen values and Eigen vectors.	1
2.8	Cholesky Decomposition	1
2.9	Eigen decomposition and Diagonalization	1
2.10	Singular Value Decomposition	1
2.11	Matrix Approximation	1
Module 3: Vector Calculus		9 hours
3.1	Differentiation of Univariate Functions - Partial Differentiation and Gradients	1
3.2	Gradients of Vector Valued Functions (Lecture 1)	1

3.3	Gradients of Vector Valued Functions (Lecture 2)	1
3.4	Gradients of Matrices	1
3.5	Useful Identities for Computing Gradients	1
3.6	Back propagation and Automatic Differentiation – Gradients in Deep Network	1
3.7	Automatic Differentiation	1
3.8	Higher Order Derivatives	1
3.9	Linearization and Multivariate TaylorSeries.	1
Module 4:Probability and Distributions		10 hours
4.1	Construction of a Probability Space	1
4.2	Discrete and Continuous Probabilities	1
4.3	Sum Rule, Product Rule	1
4.4	Bayes' Theorem.	1
4.5	Summary Statistics and Independence (Lecture 1)	1
4.6	Summary Statistics and Independence (Lecture 2)	1
4.7	Bernoulli, Binomial, Uniform (Discrete) Distributions	1
4.8	Uniform (Continuous), Poisson Distributions	1
4.9	Gaussian Distribution	1
4.10	Conjugacy and the Exponential Family (Beta – Bernoulli, Beta – Binomial Conjugacies)	1
Module 5: Optimization		7 hours
5.1	Optimization Using Gradient Descent	1
5.2	Gradient Descent With Momentum, Stochastic Gradient Descent	1
5.3	Constrained Optimization and Lagrange Multipliers (Lecture 1)	1
5.4	Constrained Optimization and Lagrange Multipliers (Lecture 2)	1
5.5	Convex Optimization	1
5.6	Linear Programming	1
5.7	Quadratic Programming	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Find the set S of all solution sinx of the following in homogeneous linear systems Ax

= b, where A and b are defined as follows: $A = \begin{bmatrix} 1 & -1 & 0 & 0 & 1 \\ 1 & 1 & 0 & -3 & 0 \\ 2 & -1 & 0 & 1 & -1 \\ -1 & 2 & 0 & -2 & -1 \end{bmatrix}$ $B = \begin{bmatrix} 3 \\ 6 \\ 5 \\ -1 \end{bmatrix}$.

2. Determine the inverses of the following matrix if possible $A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$.

Course Outcome 2 (CO 2):

1. Find the singular value decomposition (SVD) of the following matrix $\begin{bmatrix} 0 & 1 & 1 \\ \sqrt{2} & 2 & 0 \\ 0 & 1 & 1 \end{bmatrix}$.

2. Diagonalize the following matrix, if possible $\begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 1 & 0 & 0 & 3 \end{bmatrix}$.

Course Outcome 3 (CO 3):

1. Find the second order Taylor series expansion for $f(x, y) = (x + y)^2$ about (0, 0).
2. Suppose you were trying to minimize $f(x, y) = x^2 + 2y + 2y^2$. Along what vector should you travel from (5,12)?

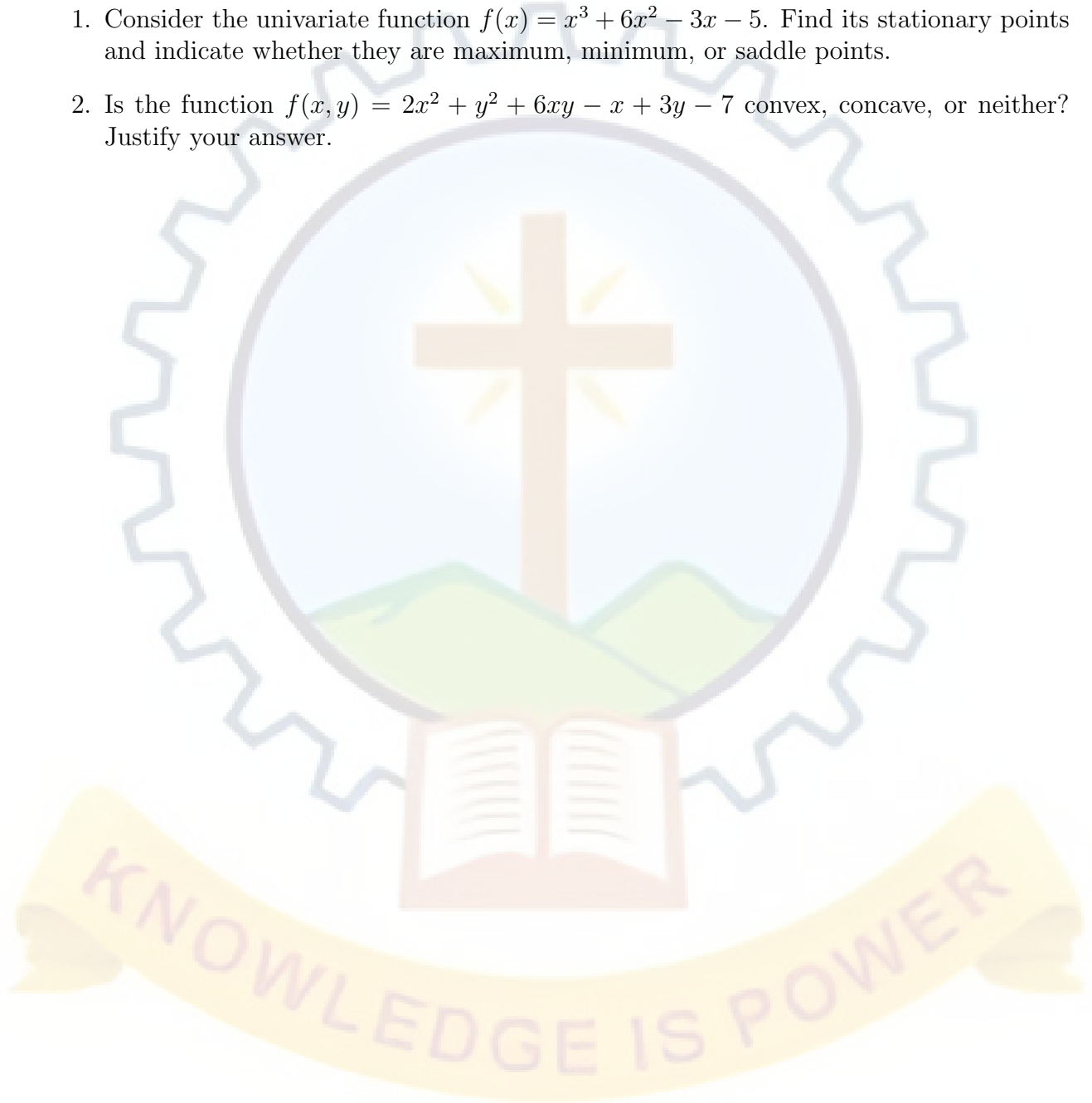
Course Outcome 4 (CO 4):

1. Let A and B be events such that $P(A)=0.45$, $P(B)=0.35$ and $P(A \cup B) = 0.5$. Find $p(A | B)$.

2. A biased coin (with probability of obtaining a head equal to $p < 0$) is tossed repeatedly and independently until the first head is observed. Compute the probability that the first head appears at an even numbered toss.

Course Outcome 5 (CO 5):

1. Consider the univariate function $f(x) = x^3 + 6x^2 - 3x - 5$. Find its stationary points and indicate whether they are maximum, minimum, or saddle points.
2. Is the function $f(x, y) = 2x^2 + y^2 + 6xy - x + 3y - 7$ convex, concave, or neither? Justify your answer.



MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM

THIRD SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2025

Course Code: B24CSM32

Course Name: MATHEMATICS FOR MACHINE LEARNING (MINOR)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Explain any 3 vector axioms.
2. Show that the linear mapping G defined as $G((x,y)) = (x + 1, y + 2)$ is not linear.
3. Find the norm of the vector $u = (2, -2, 3, -4)$.
4. Find the Eigenvalues and Eigenvectors of the 2×2 matrix $\begin{bmatrix} 4 & 2 \\ 1 & 3 \end{bmatrix}$.
5. Let $f(x, y, z) = xye^r$, where $r = x^2 + z^2 - 5$. Calculate the gradient of f at the point $(1, 3, -2)$.
6. Compute the Taylor polynomials $T_n, n = 0, \dots, 5$ of $f(x) = \sin(x) + \cos(x)$ at $x_0 = 0$.
7. Two cards are drawn in succession from a pack of 52 cards. Find the chance that the first is a king and the second a queen if the first card is (i) replaced, (ii) not replaced.
8. Show that if two events A and B are independent, then A and B are independent.
9. Explain the principle of the gradient descent algorithm.
10. Briefly explain the difference between (batch) gradient descent and stochastic gradient descent. Give an example of when you might prefer one over the other.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Show that in the space R^3 the vectors $x = (1, 1, 0)$, $y = (0, 1, 2)$, and $z = (3, 1, -4)$ are linearly dependent. (5)
- (b) A set of n linearly independent vectors in R^n forms a basis. Does the set of vectors $(2, 4, -3)$, $(0, 1, 1)$, $(0, 1, -1)$ form a basis for R^3 ? Explain your reasons. (9)

OR

12. (a) Find the rank and inverse of the matrix $A = \begin{bmatrix} 2 & -1 & 1 \\ 3 & 1 & -5 \\ 1 & 1 & 1 \end{bmatrix}$. (8)
- (b) Find a basis for the vector space V spanned by vectors $v_1 = (1, 1, 0)$, $v_2 = (0, 1, 1)$, $v_3 = (2, 3, 1)$, and $v_4 = (1, 1, 1)$. (6)
13. (a) Verify triangle inequality for $x = (1, 1, 1)$ and $y = (1, 2, 3)$ in $V_3(R)$ with standard inner product. (7)
- (b) Apply Gram-Schmidt orthogonalization process to construct an orthonormal basis for $V_3(R)$ with standard inner product for the basis $\{V_1, V_2, V_3\}$, where $V_1 = (1, 0, 1)$, $V_2 = (1, 3, 1)$ and $V_3 = (3, 2, 1)$. (7)

OR

14. (a) Compute the Singular Value Decomposition of $A = \begin{bmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{bmatrix}$. (7)
- (b) Diagonalize $A = \begin{bmatrix} 0 & 0 & -2 \\ 1 & 2 & 1 \\ 1 & 0 & 3 \end{bmatrix}$ and hence find A^{13} . (7)

15. (a) A rectangular box open at the top is to have volume 32 cubic feet. Find its dimensions if the total surface area is minimum. (7)
- (b) Find the linear approximation to the function $f(x, y) = 2 - \sin(-x - 3y)$ at the point $(0, \pi)$ and then use your answer to estimate $f(0.001, \pi)$. (7)

OR

16. (a) Discuss the maxima and minima of $xy(a - x - y)$. (5)
- (b) Compute the Taylor Series Expansion of the multivariate Function $f(x, y) = x^2 + 2xy + y^3$ at $(x_0, y_0) = (1, 2)$. (9)

17. (a) There are two bags. The first bag contains four mangoes and two apples; the second bag contains four mangoes and four apples. We also have a biased coin, which shows “heads” with probability 0.6 and “tails” with probability 0.4. If the coin shows “heads”, we pick a fruit at random from bag 1; otherwise we pick a fruit at random from bag 2. Your friend flips the coin (you cannot see the result), picks a fruit at random from the corresponding bag, and presents you a mango. What is the probability that the mango was picked from bag 2? (6)
- (b) Suppose that one has written a computer program that sometimes compiles and sometimes not (code does not change). You decide to model the apparent stochasticity (success vs. no success) x of the compiler using a Bernoulli distribution with parameter μ : $p(x | \mu) = \mu^x(1 - \mu)^{1-x}$, $x \in \{0, 1\}$. Choose a conjugate prior for the Bernoulli likelihood and compute the posterior distribution $p(\mu | x_1, \dots, x_N)$. (8)

OR

18. (a) Two dice are rolled. (6)
- A = ‘sum of two dice equals 3’
B = ‘sum of two dice equals 7’
C = ‘at least one of the dice shows a 1’
- i. What is $P(A|C)$?
ii. What is $P(B|C)$?
iii. Are A and C independent? What about B and C?
- (b) Define the probability density function and probability mass function. (8)
19. (a) Find the extrema of $f(x, y, z) = x - y + z$ subject to $g(x, y, z) = x^2 + y^2 + z^2 = 2$. (8)
- (b) Determine whether the following functions are convex function or concave. Justify your answer. (6)
- i. $f(x) = 3x^2 + 7x - 9$
ii. $f(x) = -9x^2 - x - 1$

OR

20. (a) Derive the gradient descent training rule assuming that the target function is represented as $o_d = w_0 + w_1x_1 + \dots + w_nx_n$. Define explicitly the cost/ error function E, assuming that a set of training examples D is provided, where each training example $d \in D$ is associated with the target output t_d . (8)
- (b) Find the maximum value of $f(x, y, z) = xyz$ given that $g(x, y, z) = x + y + z = 3$ and $x, y, z \geq 0$. (6)

B24CSM33	DATA COMMUNICATION CONCEPTS (Minor)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		2024

Preamble

This is a basic course in communication for awarding B. Tech. Minor in Computer Science and Engineering with specialization in Networking. The purpose of this course is to prepare learners to understand the communication entities and the associated issues in the field of Computer Science. This course covers fundamental concepts of data transmission media, digital analog transmissions, multiplexing spread spectrum, error detection correction and switching. Concepts in data communication help the learner to understand the concepts in networking and mobile communication.

Prerequisites**NIL Course Outcomes**

After the completion of the course the student will be able to:

CO 1	Describe the characteristics of signals used for Analog and Digital transmissions(Cognitive Knowledge Level: Understand)
CO 2	Discuss the features and issues in data transmission (Cognitive Knowledge Level: Understand)
CO 3	Select transmission media based on characteristics and propagation modes(Cognitive Knowledge Level: Apply)
CO 4	Use appropriate signal encoding techniques for a given scenario (Cognitive Knowledge Level: Apply)
CO 5	Explain error detection, correction techniques, and switching techniques used in data communication (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1		2					3		3
CO 2	3	3	2	3	2	3			3		3
CO 3	2	3	2	2	3	1			3		3
CO 4	2	3	2	2	3	1			3		3
CO 5	3	3	2	3	3	1			3		3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Data Transmission Basics

Communication model - Simplex, Half duplex, Full duplex transmission. Periodic Analog signals - Sine wave, Amplitude, Phase, Wavelength, Time and frequency domain, Bandwidth. Analog digital data and signals. Transmission impairments - Attenuation, Delay distortion, Noise. Data rate limits - Noiseless channel, Nyquist bandwidth, Noisy channel, Shannon's capacity formula.

MODULE 2 (10 hours)

Transmission Media

Guided Transmission Media - Twisted pair, Coaxial cable, Optical fiber. Unguided media - Radio waves, Terrestrial microwave, Satellite microwave, Infrared. Wireless Propagation - Ground wave propagation, Sky Wave propagation, Line-of-Sight (LoS) Propagation.

MODULE 3 (7 hours)

Digital Transmission and Analog Transmission

Digital data to Digital signal - Non-Return-to-Zero (NRZ), Return-to-Zero (RZ), Multilevel binary, Biphasic. Analog data to Digital signal - Sampling theorem, Pulse Code Modulation (PCM), Delta Modulation (DM). Digital data to Analog signal: Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK), Phase Shift Keying (PSK). Analog data to Analog signal: Amplitude Modulation (AM), Frequency Modulation (FM), Phase Modulation (PM).

MODULE 4 (11 hours)

Multiplexing :

Multiplexing- Space Division Multiplexing-Frequency Division Multiplexing: Wavelength Division Multiplexing - Time Division Multiplexing: Characteristics, Digital Carrier system, SONET/SDH Statistical time division multiplexing: Cable Modem - Code Division Multiplexing. Multiple Access- CDMA.

MODULE 5 (8 hours)

Error Detection, Correction and Switching:

Digital data communication techniques - Asynchronous transmission, Synchronous transmission. Detecting and correcting errors - Types of Errors, Parity check, Checksum, Cyclic Redundancy Check (CRC), Forward Error Correction (FEC), Hamming Distance, Hamming Code. Basic principles of Switching - Circuit Switching, Packet Switching, Message Switching, Asynchronous Transfer Mode(ATM)- connection-cells.

Text Books

1. Data Communications and Networking, Forouzan B. A., 6/e, McGraw-Hill, 2022.
2. Data and Computer Communication, 10/e-William Stallings, Pearson Education, Inc., 2013.

Reference Books

1. Mobile Communications-Schiller J., 2/e, Pearson Education, 2009.
2. Fundamentals of Networking and Communication -Curt M. White,7/e, Cengage learning, 2012.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Data Transmission Basics		9 hours
1.1	Introduction, Communication model - Simplex, Half duplex, Full duplex transmission	1
1.2	Periodic Analog signals - Sine wave, Amplitude, Phase, Wavelength	1
1.3	Time and frequency domain, Bandwidth	1
1.4	Analog data and signals	1
1.5	Digital data and signals	1
1.6	Transmission impairments - Attenuation, Delay distortion, Noise	1
1.7	Data rate limits - Noiseless channel, Nyquist bandwidth	1

1.8	Noisy channel, Shannon's capacity formula	2
Module 2: Transmission media		10 hours
2.1	Guided Transmission Media - Twisted pair, Coaxial cable	2
2.2	Optical fiber	1
2.3	Unguided media - Radio waves	1
2.4	Terrestrial microwave, Satellite microwave	2
2.5	Infrared	1
2.6	Wireless Propagation - Ground wave propagation	1
2.7	Wave propagation, Line-of-Sight (LoS) Propagation	2
Module 3: Digital Transmission and Analog Transmission		7 hours
3.1	Digital data to Digital signal – Non-Return-to-Zero (NRZ), Biphase	1
3.2	Return-to-Zero (RZ), Multilevel binary	1
3.3	Analog data to Digital signal - Sampling theorem	1
3.4	Pulse Code Modulation (PCM), Delta Modulation (DM)	1
3.5	Digital data to Analog signal: Amplitude Shift Keying (ASK)	1
3.6	Frequency Shift Keying (FSK), Phase Shift Keying (PSK)	1
3.7	Analog data to Analog signal: Amplitude Modulation (AM), Frequency Modulation (FM), Phase Modulation (PM)	1
Module 4: Multiplexing		11 hours
4.1	Multiplexing- Space Division Multiplexing	1
4.2	Frequency Division Multiplexing: Wavelength Division Multiplexing	2
4.3	Time Division multiplexing: Characteristics,	1
4.4	Digital Carrier system, SONET/SDH	2
4.5	Statistical time division multiplexing	2
4.6	Modem - Code Division Multiplexing	1
4.7	Multiple Access– CDMA.	1

4.8	CDMA.	1
Module 5: Error Detection, Correction and Switching		8 hours
5.1	Digital data communication techniques - Asynchronous Synchronous transmission	1
5.2	Detecting and correcting errors - Types of Errors	1
5.3	Parity check, Checksum	1
5.4	Cyclic Redundancy Check (CRC), Forward Error Correction (FEC), BCH codes	1
5.5	Reed Solomon Codes, Parity check matrix codes, Low density parity check codes, Hamming Distance, Hamming Code	1
5.6	Memory update Policies	1
5.7	Basic principles of Switching - Circuit Switching, Packet Switching, Message Switching	1
5.8	Asynchronous Transfer Mode(ATM)-connections and cells	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. What is a periodic analog signal? List the main properties of a periodic analog signal.
2. Differentiate between simplex, half-duplex, and full-duplex modes of transmission with appropriate use cases.
3. Explain how analog signals are represented in time and frequency domains. Use sine waves to illustrate amplitude, phase, and frequency.

Course Outcome 2 (CO 2):

1. What is attenuation? How can it be handled?
2. Describe the construction and characteristics of twisted-pair, coaxial cable, and optical fiber.
3. Compare radio wave, microwave, and infrared transmission with respect to range, frequency, and application.

Course Outcome 3 (CO 3):

1. How can interference be reduced using optical fiber?
2. Differentiate between NRZ, RZ, multilevel binary, and biphase encoding techniques with signal diagrams.
3. Explain the process of converting analog data to digital signal using PCM and Delta Modulation.

Course Outcome 4 (CO 4):

1. Encode the data sequence 101011100 using Multilevel binary and Biphase schemes.
2. Compare hardwired control with microprogrammed control.
3. Explain SONET/SDH architecture and its significance in digital communication systems.

Course Outcome 5 (CO 5):

1. Using Cyclic Redundancy Check (CRC), given the dataword 11110000 and the divisor 10011, show the generation of the codeword at the sender and the checking of the codeword at the receiver.
2. Illustrate the concept of Hamming code with an example and calculate the Hamming distance.
3. Compare circuit switching, packet switching, and message switching in terms of efficiency, delay, and use cases.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

THIRD SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2025

Course Code: B24CSM33

Course Name: DATA COMMUNICATION CONCEPTS (MINOR)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. What is bandwidth? Find the lowest frequency, if a periodic signal has a bandwidth of 20 Hz and the highest frequency is 60 Hz. Draw the Spectrum if the signal contains all frequencies of same amplitude.
2. Assume that a TV picture is to be transmitted over a channel with 4.5 MHz bandwidth and a 35 dB Signal-to-Noise-Ratio. Find the capacity of the channel.
3. What is the purpose of cladding in optical fibres?
4. Which wireless propagation is suitable for satellite communication? Justify your answer.
5. Explain the working of Delta Modulation with an example.
6. Illustrate the equivalent square wave pattern of the bit string 01001101 using Non-Return-to-Zero(NRZ) - Level and NRZ-Invert encoding schemes.
7. Distinguish between synchronous and statistical Time Division Multiplexing.
8. What is meant by Hamming distance.
9. Find the minimum hamming distance for the following cases:

- Detection of two errors
- Correction of two errors
- Detection of 3 errors or correction of 2 errors
- Detection of 6 errors or correction of 2 errors

10. Find the parity bit for simple even parity check for the following.

- (a) 1001010
- (b) 0001100
- (c) 1000000
- (d) 1110111

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) With the help of suitable figures, distinguish between time domain and frequency domain. (4)
- (b) Describe the different types of transmission impairments. (10)

OR

12. (a) Differentiate between Attenuation and Delay Distortion, (6)
- (b) Distinguish between Nyquist bandwidth and Shannon capacity. Consider a noiseless channel with a bandwidth of 3000 Hz transmitting a signal with
- i. Two signal levels
 - ii. Four signal levels.
- Determine the maximum bit rate in both these cases. (8)

13. (a) For a parabolic reflective antenna operating at 12 GHz with a diameter of 2 m, calculate the effective area and the antenna gain. (7)
- (b) List any four advantages and disadvantages of twisted pair, coaxial cable and fiber optic cable. (7)

OR

14. (a) Compare the features of terrestrial microwave and satellite microwave. (6)
- (b) With the help of suitable diagrams, differentiate Multi-mode and Single-mode optical fibres. How the rays are propagated in Step-index and Graded-index Multi-mode fibres. (8)
15. (a) Distinguish between data rate and signal rate. (7)
- (b) What is polar encoding? Encode the pattern 010011001110 using the two Biphasic schemes. (7)

OR

16. (a) Show the equivalent analog sine wave pattern of the bit string 010011010 using Amplitude Shift Keying, Frequency Shift Keying and Phase Shift Keying. (4)
- (b) State Sampling theorem. Explain Pulse Code Modulation with suitable figures. (10)
17. (a) Four channels are multiplexed using Time Division Multiplexing. If each channel sends 100 bytes/sec and we multiplex one byte per channel, determine the frame size, duration of a frame, frame rate and bit rate of the link. (7)
- (b) Differentiate between Synchronous TDM and Statistical TDM. Why is a statistical time division multiplexer more efficient than a synchronous time division multiplexer. (7)

OR

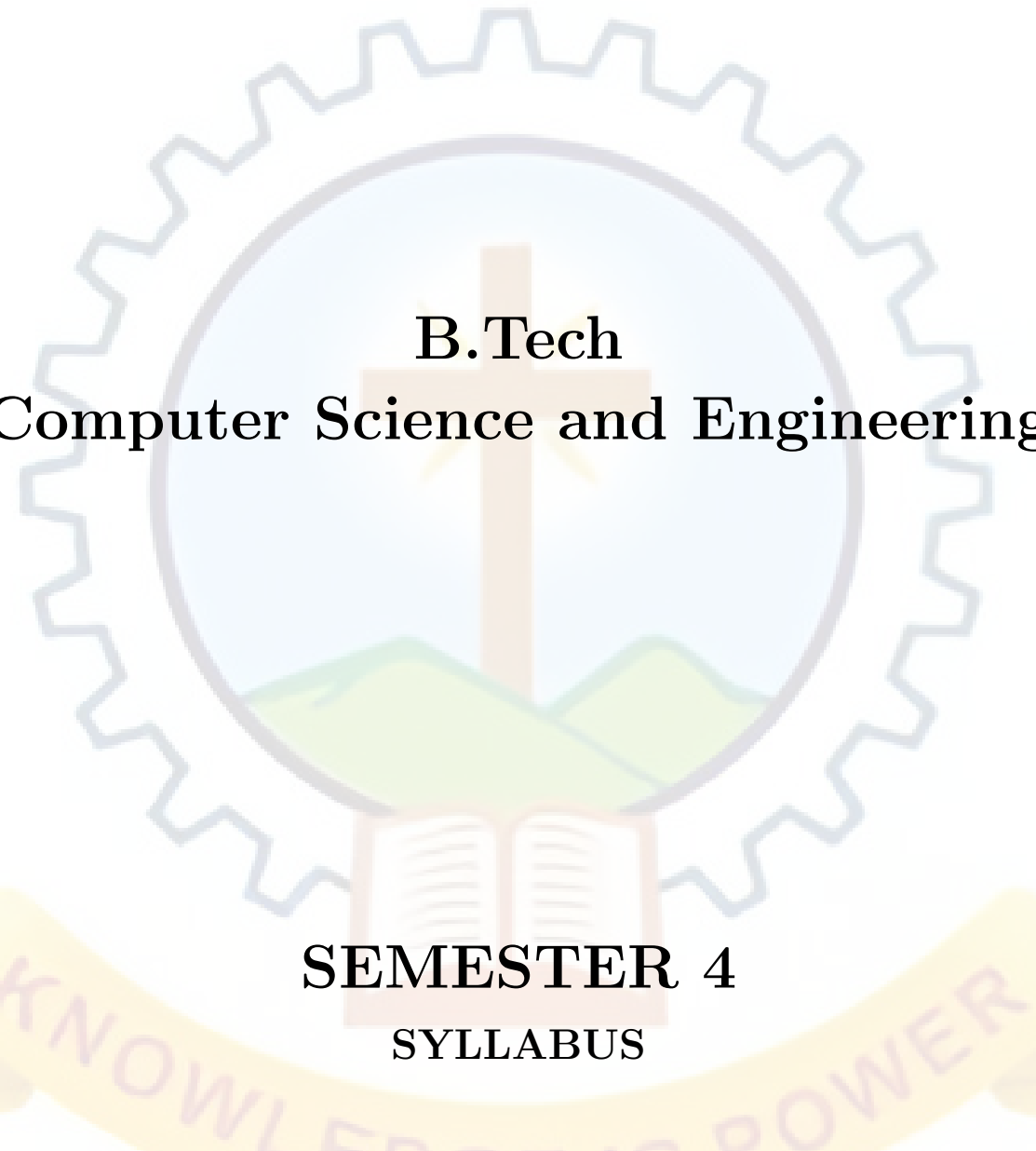
18. (a)) Explain the different techniques by which the disparity in input data rate is handled by Time Division Multiplexing. (7)
- (b) Suppose Alice and Bob are communicating using Code Division Multiple Access. Alice uses the code [+1 +1] and Bob uses the code [+1 -1]. Alice sends the data bit 0 and Bob sends the data bit 1. Show the data in the channel and how they can detect what the other person has sent. (7)
19. (a) Explain parity check with examples. (7)
- (b) Describe the need for a switch. What are the different phases in circuit switching? (7)

OR

20. (a) With the help of a suitable example, explain the virtual circuit approach of packet switching. (7)
- (b) Find the Hamming code for the data-word 1011001. Assume odd parity. (7)

MAR ATHANASIOUS COLLEGE OF ENGINEERING

Government Aided, Autonomous Institution
Kothamangalam, Kerala, India

The logo of Mar Athanasius College of Engineering is a circular emblem. It features a large gear-like outer border. Inside the circle, there is a light blue background with a central orange cross. Below the cross, there are green hills. At the bottom of the circle, there is an open book. A yellow banner with the text 'KNOWLEDGE IS POWER' in purple capital letters curves around the bottom of the emblem.

B.Tech Computer Science and Engineering

SEMESTER 4 SYLLABUS

SEMESTER 4

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24MA2T04C	GRAPH THEORY	3-1-0-5	4	4
B	B24CS2T04	OPERATING SYSTEMS	3-1-0-5	4	4
C	B24CS2T05	FORMAL LANGAUGES AND AUTOMATA THEORY	3-1-0-5	4	4
D	B24CS2T06	DATABASE MANAGEMENT SYSTEMS	3-1-0-4	4	3
E	B24HU2T03	ENTREPRENEURSHIP AND SOFTWARE MANAGEMENT SYSTEMS	2-1-0-4	3	3
F	B24CS2T07	SYSTEM SOFTWARE	2-1-0-4	3	3
G	B24CS2L05	LOGIC SYSTEM DESIGN AND OPERATING SYSTEMS LAB	0-0-3-2	3	2
H	B24CS2L06	DATABASE LAB	0-0-3-2	3	2
M	B24CSM4x	MINOR	3-1-0-5	4	4
N	B24CSH4x	HONOURS	3-1-0-5	4	4
		TOTAL		36	25

MINOR

COURSE NO.	COURSES
B24CSM41	OBJECT ORIENTED PROGRAMMING IN JAVA
B24CSM42	INTRODUCTION TO MACHINE LEARNING
B24CSM43	INTRODUCTION TO COMPUTER NETWORKS

HONOURS

COURSE NO.	COURSES
B24CSH41	ADVANCED TOPICS IN COMPUTER GRAPHICS
B24CSH42	COMPUTATIONAL FUNDAMENTALS FOR BIOINFORMATICS
B24CSH43	PRINCIPLES OF PROGRAM ANALYSIS AND VERIFICATION

B24MA2T04C	GRAPH THEORY	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

Graph Theory is a vital area of discrete mathematics with wide-ranging applications in computer science, engineering, and real-world problem solving. This course introduces students to fundamental concepts of graph theory which includes different graph structures like Eulerian and Hamiltonian graphs, directed graphs, planar graphs, connectivity concepts and matrix representations. Through both theoretical foundations and classical algorithms like Prim's, Kruskal's, Dijkstra's and Ford-Fulkerson, students will develop the skills to model, analyze, and solve complex network-based problems.

Prerequisites

Discrete Mathematical Structures (B24MA2T03C)

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Understand basic concepts of graphs, vertices and their properties, types of paths, classification of graphs and their properties. (Cognitive Knowledge Level: Understand)
CO 2	Demonstrate the fundamental theorems on Eulerian and Hamiltonian graphs. (Cognitive Knowledge Level: Understand)
CO 3	Identify equivalent definitions of trees and make use of Prim's and Kruskal's algorithms for finding minimum cost spanning tree and Dijkstra's algorithm for finding shortest paths. (Cognitive Knowledge Level: Apply)
CO 4	Utilize connectivity concepts and identify planar graphs. (Cognitive Knowledge Level: Apply)
CO 5	Explain about vector spaces and graph representations using matrices. (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1			1						
CO 2	2	1				1					
CO 3	2	1									
CO 4	2	1									
CO 5	2	1		1	1	1	1				1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 subdivisions.

SYLLABUS

MODULE 1 (8 hours)

Introduction to Graphs

Introduction- Basic definition — Application of graphs — finite, infinite and bipartite graphs — Incidence and Degree — Isolated vertex, pendant vertex and Null graph. Paths and circuits — Isomorphism, sub graphs, walks, paths and circuits, connected graphs, disconnected graphs and components.

MODULE 2 (8 hours)

Eulerian and Hamiltonian graphs

Euler graphs, Operations on graphs, Hamiltonian paths and circuits, Travelling salesman problem. Directed graphs — types of digraphs, Digraphs and binary relation, Directed paths, Fleury's algorithm.

MODULE 3 (10 hours)

Trees and Graph Algorithms

Trees — properties, pendant vertex, Distance and centres in a tree - Rooted and binary trees, counting trees, spanning trees, Prim's algorithm, Kruskal's algorithm, Dijkstra's shortest path algorithm.

MODULE 4 (11 hours)

Connectivity and Planar Graphs

Vertex Connectivity, Edge Connectivity, Cut set and Cut Vertices, Fundamental circuits, Network flows, Ford-Fulkerson algorithm, Planar graphs, Kuratowski's theorem (proof not required), Different representations of planar graphs, Euler's theorem, Geometric dual.

MODULE 5 (8 hours)

Vector spaces in Graphs and Graph Representations

Vector space associated with graphs- Sets with 1 operation, sets with 2 operations, Vectors and vector spaces, Vector space associated with a graph, Matrix representation of graphs- Adjacency matrix, Incidence Matrix, Circuit Matrix, Path Matrix.

Text Books

1. Narsingh Deo, Graph theory with Applications to Engineering and Computer Science, Dover Publications, 2016.

Reference Books

1. R. Diestel, Graph Theory, free online edition, 2016: diestel-graph-theory.com/basic.html.
2. Douglas B. West, Introduction to Graph Theory, Prentice Hall India Ltd.,2001.
3. Robin J. Wilson, Introduction to Graph Theory, Longman Group Ltd.,2010.
4. J.A. Bondy and U.S.R. Murty. Graph theory with Applications.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1:Introduction to Graphs		8 hours
1.1	Introduction- Basic definition — Application of graphs — finite and infinite graphs, bipartite graphs	1
1.2	Incidence and Degree — Isolated vertex, pendent vertex and Null graph 1	1
1.3	Paths and circuits	1
1.4	Isomorphism	1
1.5	Sub graphs, walks	1
1.6	Paths and circuits	1
1.7	Connected graphs.	1
1.8	Disconnected graphs and components	1
Module 2: Eulerian and Hamiltonian graphs		8 hours
2.1	Euler graphs	1
2.2	Operations on graphs	1
2.3	Hamiltonian paths and circuits	1

2.4	Hamiltonian paths circuits	1
2.5	Travelling salesman problem	1
2.6	Directed graphs — types of digraphs	1
2.7	Digraphs and binary relation, Directed paths	1
2.8	Fleury's algorithm	1
Module 3: Trees and Graph Algorithms		10 hours
3.1	Trees — properties	1
3.2	Trees — properties	1
3.3	Trees — properties, pendent vertex	1
3.4	Distance and centres in a tree	1
3.5	Rooted and binary tree	1
3.6	Counting trees	1
3.7	Spanning trees, Fundamental circuits	1
3.8	Prim's algorithm	1
3.9	Kruskal's algorithm	1
3.10	Dijkstra's shortest path algorithm	1
Module 4: Connectivity and Planar Graphs		11 hours
4.1	Vertex Connectivity, Edge Connectivity	1
4.2	Cut set and Cut Vertices	1
4.3	Fundamental circuits	1
4.4	Fundamental circuits	1
4.5	Network Flows	1
4.6	Ford-Fulkerson algorithm	1
4.7	Planar graphs	1
4.8	Kuratowski's theorem	1
4.9	Different representations of planar graphs	1
4.10	Euler's theorem	1
4.11	Geometric dual	1
Module 5: Vector spaces in Graphs and Graph Representations		8 hours
5.1	Sets with one operation	1
5.2	Groups of subgraphs	1

5.3	Sets with two operations	1
5.4	Finite Fields and vector spaces	1
5.5	Vector space associated with a graph.	1
5.6	Matrix representation of graphs- Adjacency matrix, Incidence Matrix	1
5.7	Circuit Matrix, Path Matrix	1
5.8	Theorems relating matrices	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

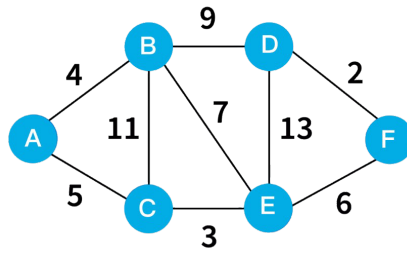
1. Differentiate a walk, path and circuit in a graph.
2. Is it possible to construct a graph with 12 vertices such that two of the vertices have degree 3 and the remaining vertices have degree 4? Justify.
3. Prove that a simple graph with n vertices must be connected if it has more than $\frac{(n-1)(n-2)}{2}$ edges.
4. Prove: If a graph (connected or disconnected) has exactly two odd degree vertices, then there must be a path joining these two vertices.

Course Outcome 2 (CO 2):

1. Define Hamiltonian circuit and Euler graph. Give one example for each.
2. Define directed graphs. Differentiate between symmetric digraphs and asymmetric digraphs.
3. Prove that a connected graph G is an Euler graph if all vertices of G are of even degree.
4. Prove: A graph G of n vertices always has a Hamiltonian path if the sum of degrees of every pair of non-adjacent vertices v_i, v_j in G satisfies the condition $d(v_i) + d(v_j) \geq n$.

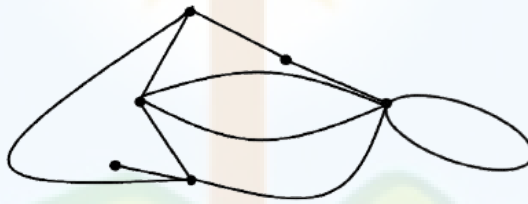
Course Outcome 3 (CO 3):

1. Prove that a tree with n vertices has $n - 1$ edges.
2. Define binary tree. Find the number of pendant vertices in a binary tree.
3. State and prove max-flow min-cut theorem.
4. Run Dijkstra's algorithm on a given graph.



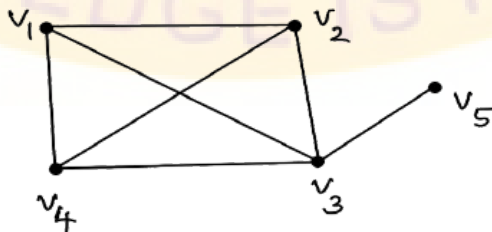
Course Outcome 4 (CO 4):

1. Define edge connectivity, vertex connectivity and separable graphs. Give an example for each.
2. Prove that a connected planar graph with n vertices and e edges has $e - n + 2$ regions.
3. Prove: Every cut set in a connected graph G must contain at least one branch of every spanning tree of G .
4. Draw the geometrical dual G^* of a given graph. Check whether G and G^* are self-duals or not, substantiate your answer clearly.



Course Outcome 5 (CO 5):

1. Show that if $A(G)$ is the incidence matrix of a connected graph G with n vertices, then the rank of $A(G)$ is $n - 1$.
2. Define a vector space.
3. Illustrate a vector space associated with a graph with an example.
4. Find the adjacency matrix of a given graph.



MODEL QUESTION PAPER

QP CODE:

Pages: 4

Reg.No.:

Name:

MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM

FOURTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2026

Course Code: B24MA2T04C

Course Name: GRAPH THEORY

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Construct a simple graph of 12 vertices with two of them having degree 1, three having degree 3 and the remaining seven having degree 10.
2. What is the largest number of vertices in a graph with 35 edges, if all vertices are of degree at least 3?
3. Define a Euler graph. Give an example of Eulerian graph which is not Hamiltonian.
4. Give an example of a strongly connected simple digraph without a directed Hamiltonian path.
5. What is the sum of the degrees of any tree of n vertices?
6. Define spanning tree with an example.
7. Show that in a simple connected planar graph G having n vertices, e edges, and no triangles $e \leq 3n - 6$.
8. Prove that edge connectivity is an upper bound for vertex connectivity.
9. Consider the circuit matrix B and incidence matrix A of a simple connected graph whose columns are arranged using the same order of edges. Prove that every row of B is orthogonal to every row of A .

10. Define a field with an example.

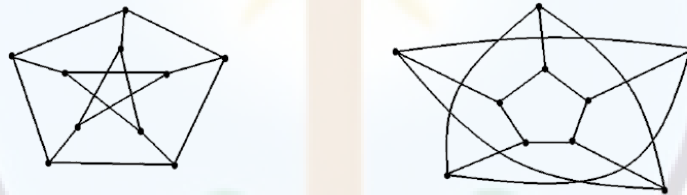
PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Prove that for any simple graph with at least two vertices, there exist two vertices of the same degree. (7)
- (b) Define subgraph, edge disjoint subgraph and vertex disjoint subgraph with example. (7)

OR

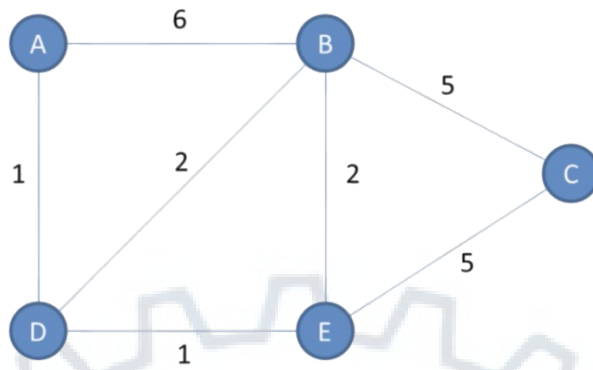
12. (a) Determine whether the following graphs are isomorphic or not. Justify your answer. (7)



- (b) Prove that a simple graph with n vertices and k components can have at most $\frac{(n-k)(n-k+1)}{2}$ edges. (7)
13. (a) Let S be a set of 5 elements. Construct a graph G whose vertices are subsets of S of size 2 and two such subsets are adjacent in G if they are disjoint. (7)
- i. Draw the graph G .
- ii. How many edges must be added to G in order for G to have a Hamiltonian cycle?
- (b) Let G be a graph with exactly two components, both being Eulerian. What is the minimum number of edges to be added to G to obtain an Eulerian graph? (7)

OR

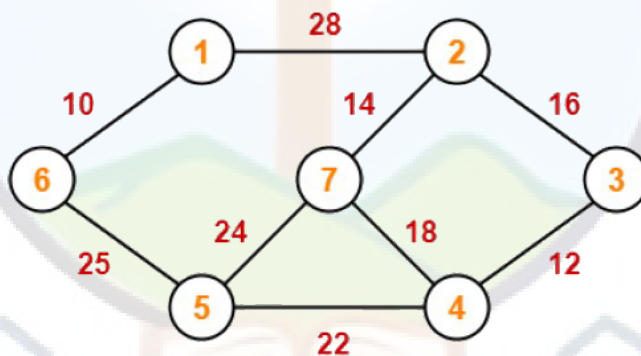
14. (a) Prove that there are $\frac{(n-1)}{2}$ edge disjoint Hamiltonian circuits in a complete graph with n vertices where $n \geq 3$. (7)
- (b) Define strongly connected, weakly connected, symmetric and asymmetric digraphs with examples. (7)
15. (a) A tree T has at least one vertex of degree 4, and at least one vertex of degree 3. Prove that T has at least 5 leaves. (7)



- (b) Use Dijkstra's algorithm to find the shortest path from vertex a to all other vertices. (7)

OR

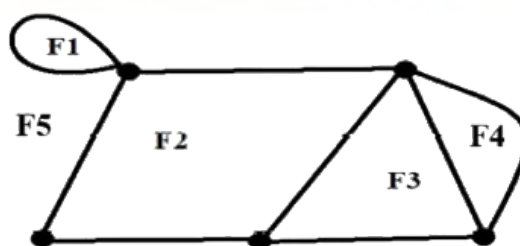
16. (a) Define pendant vertex. Prove that the number of pendant vertices in a binary tree with n vertices is $\frac{(n+1)}{2}$. (7)
 (b) Using Prim's algorithm, find the minimum spanning tree for the following weighted graph. (7)



17. (a) State and prove Euler's Theorem relating the number of faces, edges and vertices in a planar graph. (7)
 (b) Let G be a connected graph and e an edge of G . Show that e is a cut-edge if and only if e belongs to every spanning tree of G . (7)

OR

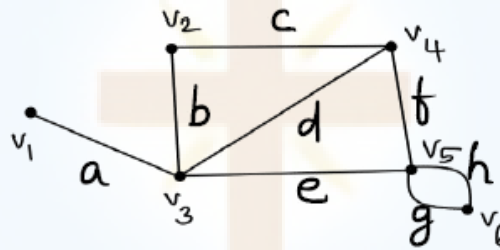
18. (a) Construct the geometric dual of the graph below. (7)



- (b) Define cut-set. Prove that every circuit in G has an even number of edges in common with any cut-set. (7)
19. (a) Draw a connected graph and show that the rank of its incidence matrix is one less than the number of vertices. (7)
- (b) Prove that the ring sum of two circuits in a graph G is either a circuit or an edge-disjoint union of circuits. (7)

OR

20. (a) Define a set with 2 operations. Give an example of a vector space associated with a graph by defining the operations. (7)
- (b) Define circuit matrix of a graph. Write the circuit matrix of following graph. (7)



B24CS2T04	OPERATING SYSTEMS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course offers students a deep understanding of how operating systems manage system resources through topics like process management, memory allocation, synchronization, and file systems, alongside real-world case studies. This course equips them to analyze, design, and optimize system-level solutions, preparing them for careers in software development, system administration, and embedded systems, as well as advanced studies in computer science.

Prerequisites

Problem Solving and Programming Techniques(A)(B24ES1T01A), Data Structures(B24CS2T01), Computer Organization And Architecture(B24CS2T02)

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the fundamental concepts of operating systems and boot processes. (Cognitive Knowledge Level: Understand)
CO 2	Demonstrate the concept of process management and process scheduling. (Cognitive Knowledge Level: Apply)
CO 3	Implement synchronization techniques to solve critical-section problems and demonstrate deadlock handling strategies. (Cognitive knowledge Level: Apply)
CO 4	Understand memory management techniques and their role in optimizing system performance. (Cognitive knowledge Level: Apply)
CO 5	Interpret the roles of file system structures and storage devices in managing system storage. (Cognitive knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	3	2	3	2	2	2			2	2
CO 2	3	3	2	3	2	2	2			2	2
CO 3	3	3	2	3	2	2	2			2	2
CO 4	3	3	2	3	2	2	2			2	2
CO 5	3	3	2	3	2	2	2			2	2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Introduction:

Operating System basic concepts - Computer System Architecture.

Case study: CPU vs GPU vs NPU.

Operating system services - System calls, user and kernel modes, Types of system calls -

Operating System structure: Simple structure, Layered approach, Microkernel, Modules - System Boot.

MODULE 2 (9 hours)

Process Management:

Process Management: Process concept: Process state, Process Control Block and Context Switch - Operations on processes-Interprocess communication: Shared-Memory Systems, Message-Passing systems – Multithreading: Benefits, Models.

Process Scheduling: Basic concepts –Schedulers, Scheduling criteria, Scheduling algorithms: First come First Served, Shortest Job First, Priority scheduling, Round-Robin scheduling, Multilevel feedback queue scheduling.

MODULE 3 (13 hours)

Process Coordination:

Process Synchronization: The Critical-Section problem-Peterson's solution-Synchronization Hardware- Mutex Locks - Semaphores -Classic problems of synchronization: Producer Consumer, Dining Philosophers and Readers-Writers.

Case study: Linux Kernel Synchronization Methods- Spin Locks

Deadlocks: Deadlock characterization - Methods for handling deadlocks - Prevention-Avoidance-Detection - Recovery from deadlock.

MODULE 4 (9 hours)

Memory management:

Address space-Swapping- Contiguous memory allocation- Segmentation- Paging: Basic method. Virtual Memory - Demand Paging- Page Replacement algorithms: FIFO, optimal, LRU, LRU-Approximation. Thrashing.

MODULE 5 (8 hours)

Storage Management:

File System: File concept: Attributes, Operations, types, structure – Access methods – File-system Mounting – Protection. File-System implementation: Overview, Partitions, Virtual file systems Magnetic Disks-Solid-State Disks - Disk Structure - Disk scheduling - Disk formatting.

Text Books

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, “Operating System Concepts”, 10th Edition, Wiley India 2018.
2. Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau Arpaci, “Operating Systems: Three Easy Pieces”, -Dusseau Books, LLC ,2017 <https://pages.cs.wisc.edu/~remzi/OSTEP/>(online version).
3. Robert Love, Linux Kernel Development, Robert Love, 3/e, 2018.

Reference Books

1. Andrew S Tanenbaum, “Modern Operating Systems”, 4th Edition, Prentice Hall, 2015.
2. William Stallings, “Operating systems”, 6th Edition, Pearson, Global Edition, 2015.
3. Garry Nutt, Nabendu Chaki, Sarmistha Neogy, “Operating Systems”, 3rd Edition, Pearson Education.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction		6 hours
1.1	Operating System basic concepts	1
1.2	Computer System Architecture. Case study: CPU vs GPU vs NPU	1
1.3	Operating system services	1
1.4	System calls, user and kernel modes, Types of system calls	1
1.5	Operating System structure: Simple structure, Layered approach, Microkernel, Modules	1
1.6	System Boot	1
Module 2: Process management		9 hours
2.1	Process concept: Process state, Process Control Block and Context Switch	1
2.2	Operations on processes	1
2.3	Interprocess communication: Shared-Memory, Message-Passing	1

2.4	Multithreading: Benefits, Models	1
2.5	Process Scheduling: Basic concepts, Schedulers, Scheduling criteria	1
2.6	Scheduling algorithms: First come First Served	1
2.7	Shortest Job First	1
2.8	Priority scheduling	1
2.9	Round-Robin scheduling, Multilevel feedback queue scheduling.	1
Module 3: Process Coordination		13 hours
3.1	Process Synchronization: The Critical-Section problem	1
3.2	Peterson's solution	1
3.3	Synchronization Hardware	1
3.4	Mutex Locks	1
3.5	Semaphores	1
3.6	Classic problems of synchronization: Producer Consumer	1
3.7	Dining Philosophers and Readers-Writers	1
Case Study: Linux Kernel Synchronization Methods - Spin Locks(1 hour)		
3.8	Deadlocks: Deadlock characterization	1
3.9	Methods for handling deadlocks: Prevention	1
3.10	Deadlock Avoidance	1
3.11	Deadlock Detection	1
3.12	Recovery from deadlock	1
Module 4: Memory Management		9 hours
4.1	Address space	1
4.2	Swapping	1
4.3	Contiguous memory allocation	1
4.4	Segmentation	1
4.5	Paging: Basic method	1
4.6	Virtual Memory – Demand Paging	1
4.7	Page Replacement algorithms: FIFO, Optimal	1
4.8	LRU, LRU-Approximation	1
4.9	Thrashing	1

Module 5: Storage Management		8 hours
5.1	File concept: Attributes, Operations, File types, structure	1
5.2	Access methods, File-system Mounting, Protection	1
5.3	File-System implementation: Overview	1
5.4	Partitions, Virtual file systems	1
5.5	Magnetic Disks	1
5.6	Solid-State Disks, Disk Structure	1
5.7	Disk scheduling: FCFS, SSTF, SCAN	1
5.8	Disk scheduling C-SCAN, LOOK, Disk formatting	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Differentiate Multiprogramming and Multitasking.
2. Illustrate the significance of bootstrap program during system boot.

Course Outcome 2 (CO 2):

1. Find the count of newly created process when the following code snippet is executed:

```
int main ()
{
for(i=0; i<3 ; i++)
fork();
return 0;
}
```

2. Consider the following process arrived at time 0 in the order P1, P2, P3, P4. Find the average waiting time and turnaround time for FCFS, SJF and RR (Q=4) scheduling algorithms.

Process	P1	P2	P3	P4
Burst Time(ms)	10	4	8	2

Course Outcome 3 (CO 3):

1. Consider a banking system that maintains an account balance with two functions: deposit (amount) and withdraw (amount). These two functions are passed the amount that is to be deposited or withdrawn from the bank account balance. Assume that two users calls the withdraw() and the deposit() concurrently. Describe how a race condition is possible and what might be done to prevent the race condition from occurring.
2. There are three process P1, P2, P3 and two resources Ra and Rb with two instances each. P1 is holding one instance of Ra, P2 is holding one instance of Rb, P3 is holding one instance of Ra and one instance of Rb. Now P1 request for Rb and P2 requests for Ra. Draw the resource allocation graph for this scenario. Identify the state of the system. Check whether there is any chance of a deadlock.

Course Outcome 4 (CO 4):

1. Given six memory partitions of 100KB, 500 KB, 200 KB, 300 KB, 600 KB (in order). How would the first-fit, best-fit and worst-fit algorithms place processes of size 212KB, 417 KB, 112KB, 426KB (in order).
2. Consider the following page reference string: 7,2,3,1,2,5,3,4,6,7,7,1,0,5,4,6,2,3,0,1. Assuming demand paging with three frames, how many page faults would occur for the following page replacement algorithms. i) LRU replacement ii) FIFO replacement iii) Optimal replacement.

Course Outcome 5 (CO 5):

1. Compare indexed allocation method with linked allocation method.
2. Assume that the read write head is at 97. The head is moving from 299 to 0. Requests are in the order 94, 82, 101, 110, 198, 75, 87, 124, 136. Find the seek distance for the following disk scheduling algorithms: i) Shortest seek time first ii) C-Scan.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FOURTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2026

Course Code: B24CS2T04

Course Name: OPERATING SYSTEMS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Describe the role of bootstrap loader in booting a computer system.
2. Identify the role of mode bit in dual mode operation of the operating system.
3. Illustrate the significance of PCB during context switch.
4. Differentiate pre-emptive and non-preemptive scheduling.
5. Examine whether mutex lock meets all the requirements of a critical section problem.
6. With an example show how a wait-for graph is used to detect deadlock
7. Consider a logical address space of 32 pages with 1024 bytes per page mapped to a physical memory of 256 frames. Calculate the size of logical address and physical address in bits.
8. Explain the role of swapping in virtual memory.
9. Define the terms i) Seek time ii) Rotational latency iii) Disk bandwidth.
10. List any three file attributes and there significance in file system.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the layered approach of OS structure. How's it different from modular approach. (7)
- (b) Explain in detail the various services of the operating system. (7)

OR

12. Describe the purpose of system call. What are the different ways to pass parameters to system call. List the basic types of system calls with examples. (14)
13. Illustrate the IPC mechanisms of shared memory and message passing. Distinguish the synchronous and asynchronous methods of message passing. (14)

OR

14. The arrival time, CPU burst and Priority of a set of processes are given in the table below. Find the average waiting time and turnaround time for SRTF, pre-emptive priority and non-preemptive priority scheduling algorithms. (14)

Process	Arrival Time	CPU Burst (ms)	Priority
P0	0	4	3
P1	2	5	2
P2	3	1	1
P3	4	3	4

15. Solve readers-writers problem using semaphore. The solution should meet all the requirements of a critical section problem. (14)

OR

16. (a) With the help of proper examples, explain how resource allocation graph can be used to avoid deadlock. (6)
- (b) Given the snapshot of system at time T0. Check whether the system is in a deadlock using deadlock detection algorithm. If P4 requests for (0, 1, 2) at time T1, determine the state of the system. (8)

Allocation			
	A	B	C
P1	0	1	2
P2	0	0	0
P3	6	3	2
P4	3	5	4

Request		
A	B	C
0	1	2
7	5	0
6	5	2
0	0	0

Available		
A	B	C
0	1	0

17. (a) Compare fixed-sized partitions and variable-sized partitions. Discuss whether memory is lost due to fragmentation in any these methods. (6)
- (b) Explain segmentation showing physical address calculation with the help of an example. What is the significance of validation bit in segmentation table? (8)

OR

18. (a) Describe how page faults are handled by the operating system. (5)
- (b) Calculate the number of page faults for the reference string 1, 2, 9, 3, 2, 5, 3, 4, 9, 1, 9, 6, 3 with three-page frames, using the i) FIFO ii) OPT iii) LRU algorithms. (9)
19. Explain the significance of file systems in secondary storage devices. How is it implemented? (14)

OR

20. Consider a disk drive that has 100 cylinders. The current position of the head is at cylinder 55. The disk queue has requests: 45, 92, 20, 63, 7, 66, 30, 35. Find the seek distance of the head for the following disk scheduling algorithms: i) SSTF ii) Elevator iii) SCAN iv) C-LOOK. Assuming that the direction of head is towards the inner tracks. (14)

B24CS2T05	FORMAL LANGAUGES AND AUTOMATA THEORY	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

Formal Languages and Automata Theory is a fundamental area of theoretical computer science that studies the mathematical structures used to define and recognize languages. This course explores different classes of formal languages, their representations using automata, and their computational properties. It covers key models such as finite automata, pushdown automata, and Turing machines, highlighting their significance in language processing and problem-solving. Understanding these concepts provides a foundation for applications in compiler design, artificial intelligence, and complexity theory.

Prerequisites

Dicsrete Mathematics(B24MA2T03C)

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Classify formal languages into regular, context-free, context-sensitive, and unrestricted categories, and design finite state automata, regular grammars, and regular expressions. (Cognitive knowledge level: Understand)
CO 2	Analyze and convert between various representations of regular languages, including regular expressions, finite automata, and regular grammars, using formal techniques like subset construction, state minimization, and the Pumping Lemma to evaluate language properties and non-regularity. (Cognitive knowledge level: Apply)
CO 3	Design, analyze, and simplify context-free grammars, construct parse trees and derivations, identify ambiguity, apply normal forms, and utilize formal techniques such as the Pumping Lemma and closure properties to determine the characteristics of context-free languages. (Cognitive knowledge level: Apply)
CO 4	Design push-down automata for context-free languages and linear bounded automata for context-sensitive grammars. (Cognitive knowledge level: Apply)
CO 5	Design and analyze Turing Machines to recognize recursive and recursively enumerable languages, apply mathematical, scientific and engineering foundations of computation, and explain the concepts of decidability, undecidability of problems, and the Halting problem. (Cognitive knowledge level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	3	2	2		1			1		2
CO 2	2	3	1	2	1				1		2
CO 3	2	3	2	2	1	1			1		2
CO 4	2	3	2	2	1	1			1		2
CO 5	3	3	2	3	3	2	2		1		3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	15	15	15
Understand	40	40	40
Apply	45	45	45
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Regular Languages and Finite Automata

Introduction to formal language theory– Alphabets, Strings, Concatenation of strings, Languages.

Chomsky classification of formal languages Need for mathematical modeling - automata, automata through simple models.

Regular Languages - Regular grammar, the formal definition of a regular expression, Building Regular Expressions, Pattern Matching.

Finite Automata- Deterministic Finite State Automata (DFA), Nondeterministic Finite State Automata (NFA), NFA with epsilon transitions, Eliminating epsilon transitions.

Applications of finite automata - text search, keyword recognition.

MODULE 2 (10 hours)

Equivalence of Regular Expression and Finite Automata

Converting Regular Expressions to NFA, Equivalence of NFAs and DFAs - The Subset Construction. DFA State Minimization (proof required), Equivalence of regular expression with finite automata, Converting FA to Regular Expressions, Equivalence of Regular grammar with FA - Conversion in both directions .

Properties of Regular Languages - Closure and Decision Properties of Regular Languages, The Pumping Lemma for Regular Languages (with formal proof), Pumping lemma as a tool to prove non-regularity of languages.

MODULE 3 (10 hours)

Context Free Languages and Context Free Grammar

Context-Free Grammars - Formal definition of a context-free grammar, Designing context-free grammars, Leftmost and Rightmost Derivations Using a Grammar, Parse Trees, Ambiguous Grammars, CFGs and programming languages.

Simplification of Context-Free Languages - Elimination of useless symbols and productions, Eliminating epsilon productions, Eliminating unit productions, Chomsky normal form, Greibach normal form.

Properties of Context-Free Languages - The Pumping Lemma for Context-Free Languages (with formal proof), Closure Properties of Context-Free Languages.

MODULE 4 (8 hours)

Pushdown Automata and Context Sensitive Languages

Pushdown Automata - Formal definition of a pushdown automaton, Empty stack and final state acceptance methods, Examples of pushdown automata, equivalence of NPDAs and CFGs - conversions in both Directions.

Context-Sensitive Languages - Context-Sensitive Grammar (CSG), Linear Bounded Automata.

MODULE 5 (8 hours)

Recursively Enumerable Languages and Turing Machines

The formal definition of a Turing machine, Examples of Turing machines, Turing machines as language acceptors, Turing machines as computers of functions, Variants of Turing Machines, Recursive and recursively enumerable languages. Universal Turing Machine and Diagonalization, Decidable and Undecidable Problems, Halting problem.

Text Books

1. John E Hopcroft, Rajeev Motwani and Jeffrey D Ullman, Introduction to Automata Theory, Languages, and Computation, 3/e, Pearson Education, 2007.
2. Dexter C. Kozen, Automata and Computability, Springer (1999).
3. Peter Linz and Susan H. Rodger, An Introduction to Formal Languages and Automata, Jones and Bartlett Publishers, Inc, 7/e, 2022.

Reference Books

1. Michael Sipser, Introduction To Theory of Computation, Cengage Publishers, 2013.
2. John C Martin, Introduction to Languages and the Theory of Computation, McGraw-Hill Education, 4/e, 2010.
3. Kavi Mahesh, Theory of Computation: A Problem-Solving Approach, Wiley, 1/e, 2012.
4. Harry R. Lewis, Christos Papadimitriou, Elements of the Theory of Computation, Pearson Education, 2/e, 2015.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Regular Languages and Finite Automata		9 hours
1.1	Introduction to formal language theory– Alphabets, Strings, Concatenation of strings, Languages.	1
1.2	Chomsky classification of formal languages	1
1.3	Need for mathematical modeling - automata, automata through simple models, automata through simple models	1
1.4	Regular Languages - Regular grammar, The formal definition of a regular expression, The formal definition of a regular expression	1
1.5	Building Regular Expressions, Pattern Matching	1
1.6	Finite Automata- Deterministic Finite State Automata (DFA)	1
1.7	Nondeterministic Finite State Automata (NFA)	1
1.8	NFA with epsilon transitions, Eliminating epsilon transitions	1
1.9	Applications of finite automata - text search, keyword recognition	1
Module 2: Equivalence of Regular Expression and Finite Automata		10 hours
2.1	Converting Regular Expressions to NFA	1
2.2	Equivalence of NFAs and DFAs - The Subset Construction	1
2.3	Equivalence of NFAs and DFAs (proof required)	1
2.4	DFA State Minimization	1
2.5	Equivalence of regular expression with finite automata	1
2.6	Converting FA to Regular Expressions	1
2.7	Equivalence of Regular grammar with FA - Conversion in both directions	1
2.8	Closure and Decision Properties of Regular Languages	1
2.9	The Pumping Lemma for Regular Languages (with formal proof)	1

2.10	Pumping lemma as a tool to prove non-regularity of languages	1
Module 3: Context Free Languages and Context Free Grammar		10 hours
3.1	Context-Free Grammars - Formal definition of a context-free grammar	1
3.2	Designing context-free grammars, Leftmost and Rightmost Derivations Using a Grammar	1
3.3	Parse Trees, Ambiguous Grammars	1
3.4	CFGs and programming languages	1
3.5	Simplification of Context-Free Languages - Elimination of useless symbols and productions,	1
3.6	Eliminating epsilon productions, Eliminating unit productions	1
3.7	CFG to Chomsky normal form	1
3.8	CFG to Greibach normal form	1
3.9	Properties of Context-Free Languages - The Pumping Lemma for Context-Free Languages (with formal proof)	1
3.10	Closure Properties of Context-Free Languages	1
Module 4: Pushdown Automata and Context Sensitive Languages		8 hours
4.1	Pushdown Automata -Formal definition of a pushdown automaton	1
4.2	Empty stack and final state acceptance methods	1
4.3	Examples of pushdown automata	2
4.4	Equivalence NPDAs and CFGs	1
4.5	conversions in both Directions	1
4.6	Context Sensitive Languages - Context Sensitive Grammar (CSG)	1
4.7	Linear Bounded Automata	1
Module 5: Recursively Enumerable Languages Turing Machines		8 hours
5.1	Turing Machines - The formal definition of a Turing machine	1
5.2	Examples of Turing machines -Turing machines as language acceptors	1

5.3	Turing machines as computers of functions	1
5.4	Variants of Turing Machines	1
5.5	Recursive and recursively enumerable languages	1
5.6	Universal Turing Machine and Diagonalization	1
5.7	Decidable and Undecidable Problems	1
5.8	Halting problem	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Given the language $L = \{w | w \in \{0, 1\}^* \text{ and } w \text{ ends with } 01\}$, construct a deterministic finite automaton (DFA) to recognize L. Explain the steps involved in designing the DFA.
2. Construct a regular expression to represent the set of all binary strings that contain at least one occurrence of the substring "101". Convert this regular expression into an equivalent finite automaton and explain the steps involved.

Course Outcome 2 (CO 2):

1. Given the regular expression $(0|1)^* 01(0|1)^*$, construct an equivalent NFA using the standard conversion process. Then, apply the subset construction method to convert the NFA into an equivalent DFA. Show all intermediate steps.
2. Minimize the number of states in the following DFA using the state minimization technique. Explain each step of the minimization process and verify that the minimized DFA is equivalent to the original. (Provide a DFA with at least 6 states for the question.)

Course Outcome 3 (CO 3):

1. Given the context-free grammar (CFG) below, construct a parse tree for the string aabb. Determine whether the grammar is ambiguous by finding multiple leftmost or rightmost derivations for the same string. If ambiguous, modify the grammar to remove ambiguity.

$$S \rightarrow aSb \mid SS \mid \epsilon$$

2. Convert the following context-free grammar (CFG) into Chomsky Normal Form (CNF). Explain each transformation step in detail.

$$S \rightarrow AB \mid aB$$

$$A \rightarrow aA \mid \epsilon$$

$$B \rightarrow bB \mid b$$

Course Outcome 4 (CO 4):

1. Design a Deterministic Pushdown Automaton (DPDA) to recognize the language

$$L = \{w\#w^R \mid w \in \{a, b\}^*\},$$

where $\#$ is a separator, and w^R denotes the reverse of w . Explain how the DPDA processes an input string.

2. Construct a Non-Deterministic Pushdown Automaton (NPDA) for the language

$$L = \{a^n b^n \mid n \geq 1\}.$$

Show all the necessary steps in the process.

Course Outcome 5 (CO 5):

1. Design a Turing Machine to compute the square of a natural number. Assume that the input is provided in unary representation.
2. Argue that it is undecidable to check whether a Turing Machine M enters a given state during the computation of a given input x .

MODEL QUESTION PAPER

QP CODE:

Pages: 4

Reg.No.:

Name:

MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM

FOURTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2026

Course Code: B24CS2T05

Course Name: FORMAL LANGAUGES AND AUTOMATA THEORY

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

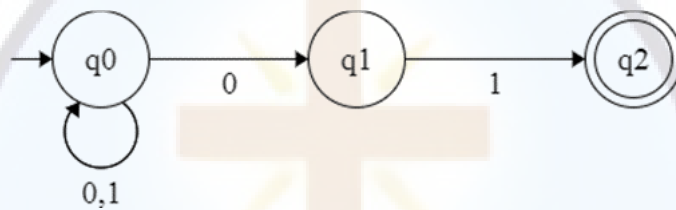
1. Construct a simple regular expression for a language containing all binary strings that end with "11".
2. Construct a finite automaton that accepts the language of all strings over a, b containing at least one occurrence of "ab".
3. Convert the regular expression $(01^*01)^*$ into an equivalent NFA using Thompson's construction. Draw the final NFA.
4. Show that the set of all strings over a, b with an even number of a's is closed under union and concatenation.
5. Define useless symbols in CFG with an example..
6. Convert the following CFG to Chomsky Normal Form (CNF)
 $S \rightarrow AB \mid a$
 $A \rightarrow BC \mid b$
 $B \rightarrow b$
 $C \rightarrow \epsilon$.
7. Design a PDA for the language of odd length binary palindromes.
8. write a Context Sensitive Grammar for the language $L = \{a^n b^n c^n \mid n \geq 1\}$ (no explanation is required, just write the set of productions in the grammar).

9. Given the string "0011", show how a Turing Machine processes it for the language $L = \{0^n 1^n \mid n \geq 1\}$.
10. Differentiate between Recursive and Recursively Enumerable Languages.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) A vending machine dispenses a product when it receives exactly 3 coins (each worth 1 unit). Design a finite automaton (DFA or NFA) to model this behavior and explain its working with state transitions. (8)
(b) Define extended transition function for an NFA. Show the processing of input $w=0101$ for the following NFA. (6)



OR

12. (a) Construct a finite automaton for recognizing valid email addresses (simplified format). Explain how the automaton differentiates between valid and invalid inputs. (8)
(b) Construct a DFA for a language that accepts all strings over 0,1 where the number of 1's is divisible by 3. Explain the approach and justify state transitions. (6)
13. (a) Convert the following regular grammar into an equivalent finite automaton:
Production Rules:
 $S \rightarrow aA \mid bB$
 $A \rightarrow aS \mid bC \mid \epsilon$
 $B \rightarrow bS \mid aC$
 $C \rightarrow aA \mid bB$ (6)
(b) Convert the finite automaton below into an equivalent regular expression: (8)
States: A, B, C
Final State: C
Transitions:

$$\delta(A, 0) = B$$

$$\delta(A, 1) = A$$

$$\delta(B, 0) = C$$

$$\delta(B, 1) = B$$

$$\delta(C, 0) = C$$

$$\delta(C, 1) = C$$

OR

14. (a) Minimize the given DFA

$$Q = \{q_0, q_1, q_2, q_3, q_4, q_5\}, F = \{q_2, q_5\} \quad (8)$$

State	0	1
q0	q1	q3
q1	q2	q4
q2	q1	q5
q3	q4	q0
q4	q5	q1
q5	q4	q2

- (b) Convert the following finite automaton into a regular expression using state elimination: (6)

- States: {A, B, C, D}
- Start State: A
- Final State: {D}
- Transitions:
 - $\delta(A, a) = B$
 - $\delta(A, b) = A$
 - $\delta(B, a) = C$
 - $\delta(B, b) = B$
 - $\delta(C, a) = C$
 - $\delta(C, b) = D$
 - $\delta(D, a) = D$
 - $\delta(D, b) = D$

15. (a) Using an example, explain ambiguity in Context Free Grammar. (6)

- (b) Convert the following CFG to Greibach Normal Form (GNF) and explain why GNF is useful. (8)

$$\begin{aligned} S &\rightarrow AB \mid a \\ B &\rightarrow bB \mid b \\ A &\rightarrow aA \mid \epsilon \end{aligned}$$

OR

16. (a) Convert the following CFG to Chomsky Normal Form (CNF) and explain each step: (7)

$$\begin{aligned} S &\rightarrow BA \mid b \\ B &\rightarrow bA \mid a \\ A &\rightarrow aB \mid \epsilon \end{aligned}$$

- (b) Design a CFG for the language $L = \{w \in \{a, b\}^* \mid w \text{ has equal number of } a\text{'s and } b\text{'s}\}$. Prove whether the CFG is ambiguous or unambiguous by deriving the string "abba". (7)

17. (a) Define a Linear Bounded Automaton (LBA). Construct an LBA for the language $L = \{0^n 1^n 2^n \mid n \geq 1\}$ and explain how it processes the string "001122". (7)
- (b) Construct an DPDA for the language

$$L = \{a^n b^{2n} \mid n \geq 1\}$$

and explain the transition process for $n = 2$. (7)

OR

18. (a) Construct a Context-Sensitive Grammar (CSG) for the language

$$L = \{w \in \{a, b\}^* \mid \text{number of a's is twice the number of b's}\}$$

Show the derivation for the string "aabb". (7)

- (b) With an example illustrate how a multi-state PDA can be transformed into an equivalent single-state PDA. (7)

19. (a) Describe the concept of a Universal Turing Machine (UTM) and explain its significance. (6)

- (b) Design a Turing Machine to accept the language $L = \{a^n b^n \mid n \geq 1\}$. Describe its transitions and explain the working of the machine with an example. (8)

OR

20. (a) Explain the concepts of recursive and recursively enumerable languages. How are they related to Turing Machines? Provide relevant examples. (6)

- (b) Explain the Halting Problem in the context of Turing machines. Why is it significant in computability theory? (8)

B24CS2T06	DATABASE MANAGEMENT SYSTEMS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	4	3	2024

Preamble

This course provides a clear understanding of fundamental principles of Database Management Systems (DBMS) with special focus on relational databases to the learners. The topics covered in this course are basic concepts of DBMS, Entity Relationship (ER) model, Relational Database principles, Relational Algebra, Structured Query Language (SQL), Physical Data Organization, Normalization and Transaction Processing Concepts. This course helps the learners to manage data efficiently by identifying suitable structures to maintain data assets of organizations and to develop applications that utilize database technologies.

Prerequisites

Data Structures(B24CS2T01), Problem Solving and Programming Techniques(A) (B24ES1T01A), Industrial Programming (B24CS1T02)

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Model conceptual database schema using ER models and explain fundamental database concepts. (Cognitive Knowledge Level: Understand)
CO 2	Analyze data organization, indexing, and relational algebra for efficient data management. (Cognitive Knowledge Level: Apply)
CO 3	Construct SQL queries to define, manipulate, and manage data within the database. (Cognitive Knowledge Level: Apply)
CO 4	Apply normalization and functional dependency principles to optimize database design and query processing. (Cognitive Knowledge Level: Apply)
CO 5	Interpret transaction management techniques, concurrency control mechanisms, and recovery methods to assess their role in ensuring database consistency and reliability; differentiate NoSQL database models and evaluate their applicability to various scenarios. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1	2	2		1	1				
CO 2	2	2		3	3						
CO 3	1	1	1	3	2						
CO 4	2	2	2	1		1					
CO 5	2	2	2	2	3	1	1				3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Introduction to Databases

Concept and Overview of Database Management Systems (DBMS) - Characteristics of Database system, Database Users, structured, semi-structured and unstructured data.

Data Models and Schema - Three Schema architecture. Database Languages, Database architectures and classification.

ER model – Entity types, entity set, attributes and keys, Relationships and constraints, cardinality, participation, notations, weak entities. Structure of Relational Databases - Integrity Constraints, Domain constraints, Entity integrity, Referential integrity constraints. Synthesizing ER diagram to relational schema.

MODULE 2 (7 hours)

Physical Data Organization, Indexing and Relational Algebra

Physical Data Organization - Review of terms: physical and logical records, blocking factor, Heap files.

Indexing and Hashing Concepts - Ordered Indices, Hash Indices, Dense and Sparse Indices, Multi Level Indices, Cluster Index, Dynamic Hashing.

Introduction to Relational Algebra - select, project, cartesian product operations, join - Equi-join, natural join.

MODULE 3 (10 hours)

Introduction to SQL Queries and Triggers

Introduction to Structured Query Language (SQL), Data Definition Language (DDL), Table definitions and operations – CREATE, DROP, ALTER, INSERT, DELETE, UPDATE.

SQL DML (Data Manipulation Language) - SQL queries on single and multiple tables, Nested queries (correlated and non-correlated) Aggregation and grouping, Views. Cursors- Implicit Cursors, Explicit cursors. Triggers- Row-level triggers, Statement-Level Triggers.

MODULE 4 (9 hours)

Database Design and Query Processing

Different anomalies in designing a database, The idea of normalization, Functional dependency, Armstrong's Axioms (proofs not required)

Closures and their computation, Equivalence of Functional Dependencies (FD), Minimal Cover (proofs not required).

First Normal Form (1NF), Second Normal Form (2NF), Third Normal Form (3NF)

Query Processing and Optimization: Steps of Query processing, Measures of Query costs.

MODULE 5 (10 hours)

Transaction Management and Emerging Trends in Databases

Transaction Processing Concepts - overview of concurrency control, Transaction Model. Significance of concurrency Control and Recovery, Transaction States, System Log, Desirable Properties of transactions.

Serial schedules, Concurrent and Serializable Schedules, Conflict equivalence and conflict serializability, Recoverable and cascade-less schedules.

Locking, Two-phase locking.

Introduction to NoSQL Concepts - types of NoSQL databases- CAP Theorem- BASE properties- Use Cases and limitations of NoSQL. Multidimensional Data Models.

Text Books

1. Fundamentals of Database Systems, Elmasri, Navathe, Pearson 7/e.
2. Making the Sense of NoSQL: A guide for Managers and rest of us, Dan McCreary and Ann Kelly, Manning, 2014.

Reference Books

1. Database System Concepts, Sliberschatz A., H. F. Korth and S. Sudarshan, Mc Graw Hill, 7/e, 2011.
2. NoSQL Data Models: Trends and Challenges (Computer Engineering: Databases and Big Data), Olivier Pivert, Wiley 2018.
3. Beginning Database Design Solutions, Rod Stephens, Wiley, 2/e, 2023.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction to Databases		9 hours
1.1	Concept and Overview of Database Management Systems (DBMS) - Characteristics of Database system, Database Users.	1
1.2	Structured, semi-structured and unstructured data. Data Models and Schema.	1

1.3	Three Schema architecture, Database Languages.	1
1.4	Database architectures and classification.	1
1.5	ER model – Entity types, entity set, attributes and keys,	1
1.6	Relationships and constraints, Cardinality, participation, notations	1
1.7	Weak entities. ER Diagram exercises	1
1.8	Structure of Relational Databases - Integrity Constraints, Domain constraints, Entity integrity, Referential integrity constraints	1
1.9	Synthesizing ER diagram to relational schema.	1
Module 2: Physical Data Organization, Indexing and Relational Algebra		7 hours
2.1	Physical Data Organization - Review of terms: physical and logical records, blocking factor, Heap files.	1
2.2	Indexing and Hashing Concepts - Ordered Indices, Hash Indices	1
2.3	Indexing and Hashing Concepts - Dense and Sparse Indices	1
2.4	Multi Level Indices, Cluster Index	1
2.5	Dynamic Hashing	1
2.6	Introduction to Relational Algebra - select, project, cartesian product operations	1
2.7	Join- Equi-join, natural join. Relational Algebra examples	1
Module 3: Introduction to SQL Queries and Triggers		10 hours
3.1	Introduction to Structured Query Language (SQL), Data Definition Language (DDL)	1
3.2	Table definitions and operations – CREATE, DROP, ALTER, INSERT, DELETE, UPDATE.	1
3.3	SQL DML (Data Manipulation Language) - SQL queries on single table.	1
3.4	SQL queries on multiple tables	1
3.5	Nested queries (correlated and non-correlated)	1
3.6	Aggregation and grouping	1
3.7	Views	1

3.8	Cursors- Implicit Cursors, Explicit cursors.	1
3.9	Triggers- Row-level triggers, Statement-Level Triggers	1
3.10	Query examples using Cursors and Triggers	1
Module 4: Database Design and Query Processing		9 hours
4.1	Different anomalies in designing a database, The idea of normalization.	1
4.2	Functional dependency, Armstrong's Axioms (proofs not required)	1
4.3	Closures and their computation	1
4.4	Equivalence of Functional Dependencies (FD)	1
4.5	Minimal Cover (proofs not required).	1
4.6	First Normal Form (1NF)	1
4.7	Second Normal Form (2NF)	1
4.8	Third Normal Form (3NF)	1
4.9	Query Processing and Optimization: Steps of Query processing, Measures of Query costs.	1
Module 5: Transaction Management and Emerging Trends in Databases		10 hours
5.1	Transaction Processing Concepts - overview of concurrency control, Transaction Model.	1
5.2	Significance of concurrency Control and Recovery, Transaction States, System Log.	1
5.3	Desirable Properties of transactions, Serial schedules	1
5.4	Concurrent and Serializable Schedules	1
5.5	Conflict equivalence and conflict serializability	1
5.6	Recoverable and cascade-less schedules.	1
5.7	Locking, Two-phase locking.	1
5.8	Introduction To NoSQL Concepts - types of NoSQL databases.	1
5.9	CAP Theorem- BASE properties. Use Cases and limitations of NoSQL	1
5.10	Multidimensional Data Models.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Draw an ER Diagram for the university management system based on the given requirements. Clearly indicate:
Entities (e.g., Student, Course, Professor, Department)
Relationships (e.g., enrollment, teaching, department assignment)
Attributes (both key attributes and descriptive attributes)
Cardinalities (one-to-many, many-to-many, etc.)
Primary keys for each entity
2. Design an ER diagram for the following scenario: There is a set of teams, each team has an ID (unique identifier), name, main stadium, and to which city this team belongs. Each team has many players, and each player belongs to one team. Each player has a number (unique identifier), name, DoB, start year, and shirt number that he uses. Teams play matches, in each match there is a host team and a guest team.

Course Outcome 2 (CO 2):

1. Consider an EMPLOYEE file with 10000 records where each record is of size 80 bytes. The file is sorted on employee number (15 bytes long), which is the primary key. Assuming un-spanned organization, block size of 512 bytes and block pointer size of 5 bytes. Compute the number of block accesses needed for retrieving an employee record based on employee number if
 - (i) No index is used
 - (ii) Multi-level primary index is used.
2. Design and implement an SQL trigger that automatically executes when a new order is inserted into the Orders table. The trigger should perform the following tasks:
Update Inventory: Deduct the order_quantity from the corresponding stock_quantity in the Books table.
Log the Transaction: Insert a new record into the OrderHistory table capturing the order_id, book_id, current date as order_date, the stock_quantity before the update (old_stock), and the updated stock_quantity (new_stock).

Course Outcome 3 (CO 3):

1. For the SQL query, SELECT A, B FROM R WHERE B='apple' AND C = 'orange' on the table R(A, B, C, D), where A is a key, write any three equivalent relational algebra expressions.

2. In the following tables, foreign keys have the same name as primary keys except DIRECTED-BY, which refers to the primary key ARTIST-ID. Consider only single-director movies. MOVIES(MOVIE-ID, MNAME, GENRE, LENGTH, DIRECTED-BY) ARTIST(ARTIST-ID, ANAME) ACTING(ARTIST-ID, MOVIE-ID) Write SQL expressions for the following queries:
- Name(s) and director name(s) of movie(s) acted by 'Jenny'.
 - Names of actors who have never acted with 'Rony'
 - Count of movies genre-wise.
 - Name(s) of movies with maximum length

Course Outcome 4 (CO 4):

- Given the FDs $P \rightarrow Q$, $P \rightarrow R$, $QR \rightarrow S$, $Q \rightarrow T$, $QR \rightarrow U$, $PR \rightarrow U$, write the sequence of Armstrong's Axioms needed to arrive at the following FDs: (a) $P \rightarrow T$ (b) $PR \rightarrow S$ (c) $QR \rightarrow SU$
- Consider a relation PLAYER (PLAYER-NO, PLAYER-NAME, PLAYER-POSN, TEAM, TEAM-COLOR, COACH-NO, COACH-NAME, TEAM-CAPTAIN). Assume that PLAYER-NO is the only key of the relation and that the following dependencies hold:
 $TEAM \rightarrow \{TEAM-COLOR, COACH-NO, TEAM-CAPTAIN\}$
 $COACH-NO \rightarrow COACH-NAME$.
 - Is the relation in 2NF? If not, decompose to 2NF.
 - Is the relation in 3NF? If not, decompose to 3NF.

Course Outcome 5 (CO 5):

- Describe the salient features of NoSQL databases. Give example of a document in MongoDB.
- Determine if the following schedule is *recoverable*. Is the schedule *cascade-less*? Justify your answer. $r1(X)$, $r2(Z)$, $r1(Z)$, $r3(X)$, $r3(Y)$, $w1(X)$, $c1$, $w3(Y)$, $c3$, $r2(Y)$, $w2(Z)$, $w2(Y)$, $c2$.

MODEL QUESTION PAPER

QP CODE:

Pages: 4

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FOURTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2026

Course Code: B24CS2T06

Course Name: DATABASE MANAGEMENT SYSTEMS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Differentiate between Super key, Candidate key and Primary key.
2. Describe the concept of a weak entity used in data modelling. Define the terms owner entity type, identifying relationship type.
3. Explain any three differences between dense index and sparse index
4. Differentiate between theta join and natural join operations.
5. Compare DDL and DML with the help of an example.
6. Illustrate the concept of trigger in SQL with an example.
7. Cite the Armstrong's axioms of FD.
8. Describe First Normal Form (1NF), Second Normal Form (2NF), and Third Normal Form (3NF) with appropriate examples.
9. Describe log-based recovery in DBMS and its importance with an example.
10. Summarize the features of NoSQL databases.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Draw an ER diagram based on the following information: (7)

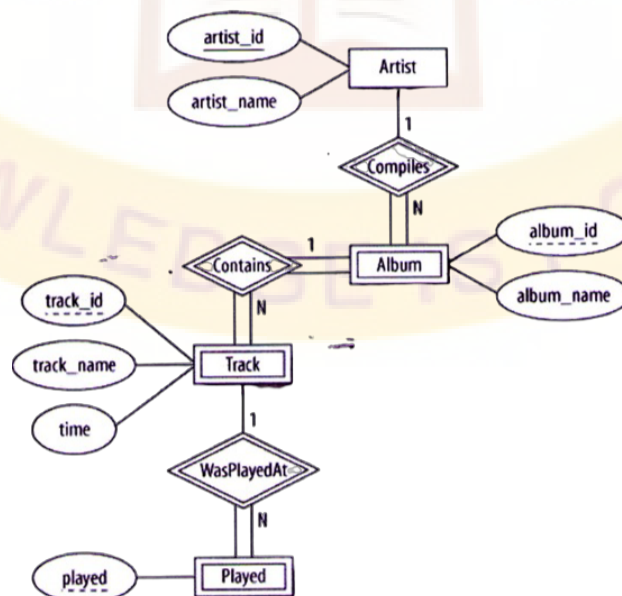
- Manufacturers have a name, which we may assume is unique, an address, and a phone number.
- Products have a model number and a type. Each product is made by one manufacturer and different manufacturers may have different products with the same model number. However, you may assume that no manufacturer would have two products with the same model number.
- Customers are identified by their unique social security number. They have email addresses, and physical addresses. Several customers may live at the same(physical) address, but we assume that no two customers have the same email address.
- An order has a unique order number, and a date. An order is placed by one customer. For each order, there are one or more products ordered, and there is a quantity for each product on the order.

(b) Explain the difference between logical data independence and physical data independence in a database system. Illustrate each concept with a real-world example. (7)

OR

12. (a) Describe any three types of database end users. Explain their roles and provide examples of how they interact with a database system. (7)

(b) Interpret the following ER diagram: (7)



13. (a) With the help of an example, explain Single-level indexing and multi-level indexing. Explain the difference (any four) between single level indexing and multi-level indexing. (8)
- (b) Explain a situation where a multi-level index would be significantly less effective than a single-level index, and vice versa. (6)

OR

14. (a) Consider the following schema. (8)
- Suppliers(sid(pk), sname, address)
Parts(pid(pk), pname, color)
Catalog(sid,pid(pk), cost)
- The primary key fields are underlined; Foreign keys have the same name as primary keys. Assume integer domain for the attributes sid, pid, cost and string domain for the attributes sname, address, pname, color. Write relational algebra expressions for the given questions. (Use * symbol for natural join and symbol for join)
- Find the names of suppliers who supply some red part.
 - Find the sids of suppliers who supply some red or green part.
 - Find the sids of suppliers who supply every red part or supply every green part.
 - Find the pids of parts supplied by at least two different suppliers
- (b) Explain the left outer join, right outer join and full outer join operations with suitable example (6)
15. (a) A company has two tables: Sales (SaleID - Primary Key, ProductID - Foreign Key, QuantitySold, SaleDate) and Inventory (ProductID - Primary Key, ProductName, StockQuantity, Price). Write an SQL trigger "UpdateStockAfterSale" to automatically update StockQuantity in the Inventory table when a new sale is recorded in Sales. Explain its working with an example, including the initial Inventory state, an INSERT into Sales, and the updated Inventory table. (7)
- (b) A company maintains an Employee database with the following table: Employee (EmpID, Name, Department, Salary, JoiningDate). Write SQL Queries to perform the following operations: (7)
- Insert a new employee record into the table.
 - Update the salary of an employee based on their EmpID.
 - Delete an employee record who has left the company.
 - Retrieve the names and salaries of employees working in the "IT" department.

OR

16. (a) Write SQL DDL statements based on the following database schema (Assume suitable domain types): (8)
- Employee (eid(pk), name, designation, salary, compid)
Company (compid(pk),cname, address, turnover)

- i. Create the above mentioned tables assuming each company has many employees. Mention the primary key, foreign key and not null constraints.
 - ii. Insert values into both the tables. Mention in which order insertions will be carried out.
 - iii. Modify the table Employee to include a new column “years-of-exp”
 - iv. Increment the salary of employees whose salary is less than Rs 25000 by 5%.
- (b) Illustrate any three ways of using INSERT statement in SQL. (6)
17. (a) Given a relation $R(A, B, C, D, E)$ with the functional dependencies $A \rightarrow B$, $B \rightarrow C$, $CD \rightarrow E$, find the closure of A and explain its significance in determining the keys of the relation. (6)
- (b) Describe the different steps of query processing and explain their significance. How are query cost measures used to evaluate query performance? (8)

OR

18. (a) P and Q are two set of FDs for a relational schema $R(A, B, C, D)$.
 $p = \{A \rightarrow B, B \rightarrow C, C \rightarrow D\}$ and $Q = A \rightarrow BC, C \rightarrow D$. Check whether P covers Q and Q covers P? Also check whether P and Q are equivalent. (6)
- (b) Consider a relation R with five attributes (A, B, C, D, E). You are given the following dependencies: $A \rightarrow B$, $BC \rightarrow E$, and $ED \rightarrow A$.
 - i. List all keys for R.
 - ii. Is R in 3NF?
 - iii. Convert the relation to the next higher normal form. (8)
19. (a) Determine if the following schedule is recoverable. Is the schedule cascadeless? Justify your answer. (7)
 $rl(X), r2(Z), rl(Z), r3(X), r3(Y), wl(X), cl, w3(Y), c3, r2(Y), w2(Z), w2(Y), c2$
(Note: $ri(X)/wi(X)$ means transaction T_i issues read/write on item X; ci means transaction T_i commits.)
- (b) Check whether the following schedule is conflict serializable or not and find an equivalent serial schedule if possible.
 $rl(X), r2(Z), rl(Z), r3(X), r3(Y), wl(X), w3(Y), r2(Y), w2(Z), w2(Y)$
(Note: $ri(X)/wi(X)$ means transaction T_i issues read/write on item X) (7)

OR

20. (a) Describe the different states of a transaction with the help of a neat sketch. (7)
- (b) Differentiate serial and concurrent schedules. Elaborate conflict serializability with suitable example. (7)

B24HU2T03	ENTREPRENEURSHIP AND SOFTWARE MANAGEMENT SYSTEMS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course aims to learn the techniques to effectively plan, manage, execute, and control projects within time and cost targets with a focus on Information Technology and Service Sector. This course helps to learn fundamentals of Agile methodology, agile principles and agile project management techniques such as Scrum, XP and incremental design.

Prerequisites

Nil

Course Outcomes

After the completion of the course, the student will be able to:

CO 1	Understand the fundamentals of digital business models, startup financials, modern tech marketing, and AI-driven decision-making in the tech startups. (Cognitive Knowledge Level: Understand)
CO 2	Recognize the significance of engaging with business stakeholders to identify and gather requirements for a software system. (Cognitive Knowledge Level: Understand)
CO 3	Implement Agile methodologies throughout the Software Development Life Cycle. (Cognitive Knowledge Level: Apply)
CO 4	Apply various SCRUM practices in project management. (Cognitive Knowledge Level: Apply)
CO 5	Integrate Extreme Programming (XP) methods and incremental design principles into Agile software design and development. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2		3	3	1			2	2	3
CO 2	1	1	2	1		1	3	1	2		
CO 3	1	3	3	3	3	2		3	2	2	3
CO 4	1	1		1	1	1		3	1	2	
CO 5	1	2	2	2	3	2					2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Entrepreneurship, Business Design, Marketing and Startup Pitching

Entrepreneurial Mindset-SimVenture, GoVenture; Identifying Business Opportunities- Google Trends, SurveyMonkey; Innovation and Business Model Tools- Canvanizer, Miro; Digital Business Models- Shopify, Chargebee; Business Model Canvas-Miro, Lucidchart; Startup Financials and Pitching for Tech Startups-LivePlan, Pitch; Modern Tech Marketing and AI-Driven Decisions- HubSpot, Marketo, Google Analytics, Hotjar, IBM Watson.

MODULE 2 (6 hours)

Introduction to Software Engineering and Agile Software

Introduction to Software Engineering - Professional software development - Git and GitHub/GitLab; Software engineering ethics- OWASP ZAP. Software process models - The waterfall model, Incremental development - Jira, Trello, Lucidchart/ Draw.io. Process activities - Software specification - Figma, UI/UX prototyping, Jira; Software design and implementation - Visual Studio Code, Eclipse, Git, GitHub; Software validation-Cypress(web testing), JUnit, Pytest (unit testing), Postman (API testing); Software evolution-VS Code with Git integration, CI tools like CircleCI; Coping with change - Prototyping, Incremental delivery, Boehm's Spiral Model-Adobe XD/Figma, Azure DevOps/GitLab; Agile software development - Agile methods, agile manifesto - values and principles- Scrum, Kanban, XP-Roles like Product Owner, Scrum Master, Dev Team- Agile tools like Jira, Azure Boards, Miro.

MODULE 3 (6 hours)

Agile Development and project management

Agile Management – Agile Software Development-Jira, Azure Boards; Traditional Model vs. Agile Model-Lucidchart, Draw.io, Miro; Classification of Agile Methods-Scrum, Kanban, XP;Agile Manifesto and Principles-Miro, Jamboard; Agile Project Management – Agile Team Interactions-Microsoft Teams, Zoom, Miro, FigJam; Ethics in Agile Teams-Poll Everywhere;Agility in Design-Figma, Adobe XD; Testing-Cypress, PyTest, Postman; Agile Documentations-Google Docs; Agile Drivers, Capabilities and Value-Jira Reports, EazyBI, Google Data Studio, Power BI.

MODULE 4 (8 hours)

Introduction to SCRUM practices in project management

Agile Scrum Framework-Jira Scrum Boards, Azure DevOps; Scrum Artifacts-Jira, Trello; Meetings-Zoom, Microsoft Teams, Miro, FigJam; Activities and Roles-GitHub

Projects; Scrum Team Simulation-ScrumSim, Agile Lego Simulation Kits; Scrum Planning Principles-Planning Poker Online, Miro templates for estimation, Agile Cards; Product and Release Planning-Product Plan(Linear); Sprinting: Planning, Execution, Review and Retrospective-Jira Sprint Boards, Burndown plugins; User story definition and Characteristics- UserStoryMap.io, StoriesOnBoard, Jira User Story Templates, Acceptance tests and Verifying stories-Cucumber, SpecFlow, TestRail, Postman; Burn down chart-Jira Agile Reports, EazyBI, Excel/Google Sheets; Daily scrum- Slack with Geekbot, Microsoft Teams Check-ins; Scrum Case Study-Jira + GitHub, Confluence + Miro, Notion + Trello, LMS platforms like Moodle, Google Classroom.

MODULE 5 (10 hours)

Agile software design and development:

Agile design practices- Figma, Sketch, Adobe XD, Lucidchart, Draw.io, Miro; Role of design Principles; Need and significance of Refactoring-IntelliJ IDEA, VS Code; Refactoring Techniques-ReSharper, Eclipse Refactoring Tools, PyCharm Code Actions; Continuous Integration-GitHub Actions, GitLab CI/CD, CircleCI; Automated build tools-npm, Webpack; Version control-Git, GitHub, GitLab; Agility and Quality Assurance: Agile Interaction Design-Figma, Marvel App, UXPin; Agile approach to Quality Assurance-Zephyr, Allure, Xray for Jira; Test Driven Development-JUnit, NUnit, RSpec, PyTest, Jest; Pair programming: Issues and Challenges-Visual Studio Live Share, CodeTogether; Introduction to Extreme Programming (XP): XP Life cycle-Jira XP Templates, Agilefant; The XP Team, XP Concepts-SonarQube, CodeScene, Refactoring; Technical Debt-SonarCloud, CodeClimate; Timeboxing, Stories, Velocity-Jira Velocity Reports, Burndown Charts, Agile Cards, Scrumpy Poker; Adopting XP: Pre-requisites, Challenges-Atlassian suite (Jira, Bitbucket),Scrum wise, XP-focused Trello boards, Agile Maturity Assessment tools.

Text Books

1. Peter F. Drucker, Innovation and Entrepreneurship, HarperBusiness, 2006.
2. Alexander Osterwalder and Yves Pigneur, Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers, Wiley, 2010.
3. Mike Cohn, Succeeding with Agile: Software Development Using Scrum, Addison-Wesley, 1/e, 2009.
4. David J. Anderson and Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall , 2003.
5. Hazza and Dubinsky, Agile Software Engineering, Series: Undergraduates Topics in Science ,Springer, 2009.
6. Robert C. Martin, Agile Software Development- Principles, Patterns and Practices, Prentice Hall, 2013.
7. Ken Schawber, Mike Beedle, Agile Software Development with Scrum, Pearson, 2001.

8. Craig Larman, Agile and Iterative Development: A manager's Guide, Addison-Wesley, 2004.

Reference Books

1. Heidi M. Neck, Christopher P. Neck & Emma L. Murray, Entrepreneurship: The Practice and Mindset, SAGE Publications Inc., 3rd Edition, 2023.
2. Alejandro Cremades, The Art of Startup Fundraising: Pitching Investors, Negotiating the Deal, and Everything Else Entrepreneurs Need to Know, Wiley, 2023.
3. Roman Pichler, Agile Product Management with Scrum, Addison-Wesley, 1/e, 2010.
4. Kenneth S. Rubin, Essential Scrum: A Practical Guide to the Most Popular Agile Process, Addison Wesley, 2012.
5. James Shore and Shane Warden, The Art of Agile Development, O'Reilly Media, 2007.
6. Cohn, Mike, User Stories Applied: For Agile Software Development, Addison Wisley, 2004.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Entrepreneurship, Business Design, Marketing and Startup Pitching		6 hours
1.1	Entrepreneurial Mindset, Identifying Business Opportunities	1
1.2	Innovation and Business Model Tools, Digital Business Models	1
1.3	Business Model Canvas	1
1.4	Startup Financials and Pitching for Tech Startups	1
1.5	Modern Tech Marketing	1
1.6	AI-Driven Decisions	1
Module 2: Introduction to Software Engineering and Agile Software		6 hours
2.1	Introduction to Software Engineering, Professional software development, Software engineering ethics	1

2.2	Software process models - The waterfall model, Incremental development	1
2.3	Process activities - Software specification, Software design and implementation	1
2.4	Software validation, Software evolution.	1
2.5	Coping with change - Prototyping, Incremental delivery, Boehm's Spiral Model.	1
2.6	Agile software development - Agile methods, agile manifesto - values and principles	1
Module 3: Agile Development and project management		6 hours
3.1	Agile Management and Agile Software Development	1
3.2	Traditional Model vs. Agile Model	1
3.3	Classification of Agile Methods, Agile Manifesto and Principles, Agile Project Management	1
3.4	Agile Team Interactions, Ethics in Agile Teams	1
3.5	Agility in Design, Testing, Agile Documentations	1
3.6	Agile Drivers, Capabilities and Value.	1
Module 4: Introduction to SCRUM practices in project management		8 hours
4.1	Agile Scrum Framework, Scrum Artifacts, Meetings, Activities and Roles	1
4.2	Scrum Team Simulation, Scrum Planning Principles, Product and Release Planning (Lecture 1)	1
4.3	Scrum Team Simulation, Scrum Planning Principles, Product and Release Planning (Lecture 2)	1
4.4	Sprinting: Planning, Execution, Review and Retrospective	1
4.5	User story definition and Characteristics	1
4.6	Acceptance tests and Verifying stories	1
4.7	Burn down chart, Daily scrum	1
Case Study: Scrum (1 hour)		
Module 5: Agile software design and development		10 hours
5.1	Agile design practices, Role of design Principles	1
5.2	Need and significance of Refactoring, Refactoring Technique	1

5.3	Continuous Integration, Automated build tools, Version control	1
5.4	Agility and Quality Assurance: Agile Interaction Design, Agile approach to Quality Assurance, Test Driven Development (Lecture 1)	1
5.5	Agility and Quality Assurance: Agile Interaction Design, Agile approach to Quality Assurance, Test Driven Development (Lecture 2)	1
5.6	Pair programming: Issues and Challenges	1
5.7	Introduction to Extreme Programming (XP): XP Life cycle, The XP Team, XP Concepts	1
5.8	Refactoring, Technical Debt	1
5.9	Time boxing, Stories, Velocity	1
5.10	Adopting XP: Pre-requisites, Challenges	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Define the term Entrepreneurial Mindset. List any four characteristics of a successful entrepreneur.
2. Explain how a Business Model Canvas helps in identifying and analyzing a new business opportunity.
3. Apply any Innovation Tool (e.g., SWOT analysis, Design Thinking) to identify a business opportunity in the Digital Business sector.
4. Analyze the components of a successful Startup Pitch and evaluate how AI-driven decisions can enhance Modern Tech Marketing strategies.
5. Design a Business Model Canvas for a technology startup of your choice. Include key elements like Value Proposition, Customer Segments, Revenue Streams, and Cost Structure.

Course Outcome 2 (CO 2):

1. List the core values of the Agile Manifesto and explain their significance in Agile software development.

2. Explain the key differences between the Waterfall model and Incremental development in terms of their software process flow.
3. Apply the Boehm's Spiral Model to a real-world software development scenario and describe how risk analysis is integrated at each phase.
4. Analyze how prototyping supports software specification and validation activities in the development process.

Course Outcome 3 (CO 3):

1. List any four principles of the Agile Manifesto and briefly describe their purpose..
2. Compare and contrast the Traditional Software Development Model with the Agile Model in terms of flexibility, team structure, and documentation.
3. Explain the classification of Agile methods and give one example for each category.
4. Demonstrate how Agile team interactions influence project outcomes in a real-world software development scenario.

Course Outcome 4 (CO 4):

1. Define a user story and list its main characteristics in the Scrum framework.
2. Explain the roles and responsibilities of the Scrum Master, Product Owner, and Development Team in a Scrum project.
3. Illustrate the process of a typical Sprint cycle by describing its key phases: planning, execution, review, and retrospective.
4. Describe the purpose and usage of a burn down chart. How does it help in monitoring project progress?

Course Outcome 5 (CO 5):

1. What is refactoring in agile design, and why is it significant in software development?
2. Explain how Test-Driven Development (TDD) contributes to Agile quality assurance.
3. Demonstrate the use of version control and continuous integration tools in managing collaborative code development.
4. Describe the XP lifecycle and the role of "stories" and "velocity" in planning and tracking XP projects.
5. Analyze the challenges and potential benefits of adopting Pair Programming in an Agile team.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FOURTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2026

Course Code: B24HU2T03

***Course Name: ENTREPRENEURSHIP AND SOFTWARE
MANAGEMENT SYSTEMS***

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Differentiate between Traditional Business Models and Digital Business Models with suitable examples.
2. Explain the importance of Identifying Business Opportunities in entrepreneurship.
3. How does Software Validation help in software development?
4. Why is continuous customer collaboration emphasized in Agile development?
5. Compare the traditional software development model with the Agile model in terms of flexibility, customer involvement, and delivery approach.
6. Explain any three principles of the Agile Manifesto and how they contribute to effective software development.
7. What are the three main Scrum Artifacts, and what is their purpose?
8. Briefly explain the importance of the Daily Scrum meeting in Agile Scrum.
9. What is Refactoring, and why is it important in Agile development?
10. Briefly explain the concept of Test-Driven Development (TDD) and its benefits in Agile.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the concept of Entrepreneurial Mindset. Discuss how developing an entrepreneurial mindset helps in identifying and evaluating business opportunities in the digital era. (7)
- (b) Analyze the role of Innovation Tools in developing a competitive business model. Illustrate how a startup can use any two innovation tools to improve its business model. (7)

OR

12. (a) Apply the Business Model Canvas framework to develop a business idea for a technology-based startup. Explain each component briefly with respect to your idea. (7)
- (b) Design a marketing strategy for a tech startup using AI-driven decision-making tools. Evaluate how modern tech marketing approaches help startups achieve competitive advantage. (7)
13. (a) Explain the role of Professional Software Development and discuss the key challenges faced by software engineers. (7)
- (b) Compare and contrast the Waterfall Model and Incremental Development in terms of flexibility, risk handling, and efficiency. (7)

OR

14. (a) Describe the key phases of Software Specification, Design, Implementation, Validation, and Evolution with real-world examples. (7)
- (b) Explain how Prototyping and Incremental Delivery help in handling changing customer requirements. (7)
15. (a) Compare and contrast the Traditional Software Development Model with the Agile Model. (7)
- (b) Explain the Agile Manifesto and its core principles with suitable examples. (7)

OR

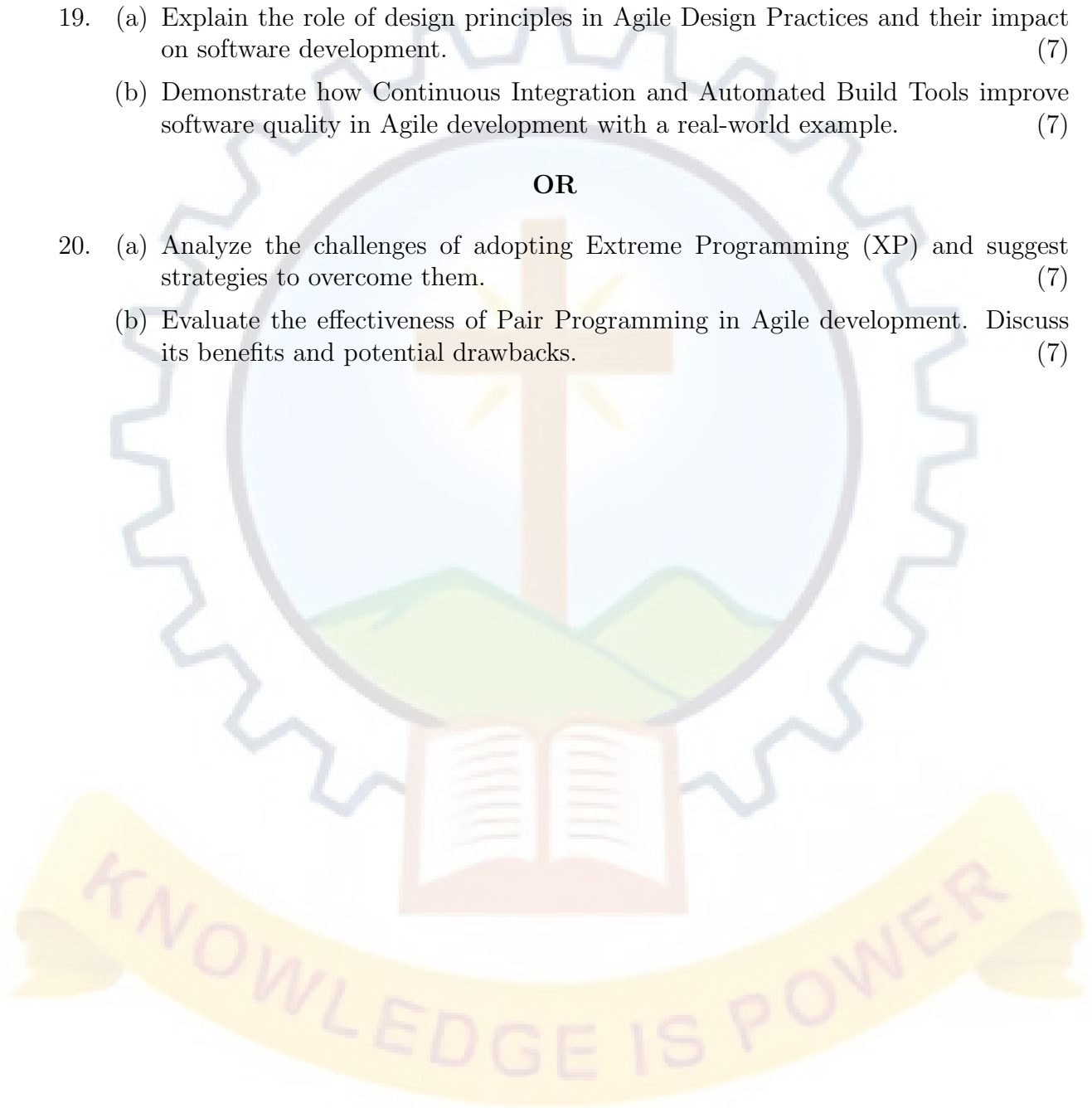
16. (a) Discuss the significance of Agile Team Interactions and Ethics in Agile Teams. (7)
- (b) How do Agile Drivers, Capabilities, and Value contribute to Agile Project Management? (7)
17. (a) Explain the key roles in the Agile Scrum Framework and their responsibilities. (7)
- (b) Discuss the importance of Sprint Planning, Execution, Review, and Retrospective in Scrum. (7)

OR

18. (a) What is a Burndown Chart? Explain its significance in tracking project progress in Scrum. (7)
- (b) Describe the characteristics of a good User Story and explain how Acceptance Tests help in verifying stories. (7)
19. (a) Explain the role of design principles in Agile Design Practices and their impact on software development. (7)
- (b) Demonstrate how Continuous Integration and Automated Build Tools improve software quality in Agile development with a real-world example. (7)

OR

20. (a) Analyze the challenges of adopting Extreme Programming (XP) and suggest strategies to overcome them. (7)
- (b) Evaluate the effectiveness of Pair Programming in Agile development. Discuss its benefits and potential drawbacks. (7)



B24CS2T07	SYSTEM SOFTWARE	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4		2024

Preamble

This course aims to create awareness about the low-level codes very close to the hardware and the environment where programs can be developed and executed. This course helps the learner to understand the machine-dependent and machine-independent system software features and to design/implement system software like assembler, loader, linker, microprocessor, and device drivers. Study of system software develops the ability to design interfaces between software applications and computer hardware.

Prerequisites

Data Structures (B24CST01) and Computer Organization and Architecture(B24CS2T02)

Course Outcomes

After the completion of the course, the student will be able to:

CO 1	Distinguish software into system and application software categories. (Cognitive Knowledge Level: Understand)
CO 2	Outline the basic and advanced architectural features present in machines. (Cognitive Knowledge Level: Apply)
CO 3	Classify the system software functionalities that rely on machine-specific characteristics. (Cognitive Knowledge Level: Apply)
CO 4	Discuss the aspects of system software that operate independently of the machine architecture. (Cognitive Knowledge Level: Understand)
CO 5	Design algorithms for system software and analyze the effect of data structures used. Understand the features of device drivers, editing, and debugging tools. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	3			1						
CO 2	2	3	1								
CO 3	2	3	1								
CO 4	2	3									
CO 5	3	3	2	3	3						

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (5 hours)

Introduction to System Software:

System Software vs Application Software, Different System Software– Assembler, Linker, Loader, Macro Processor, Text Editor, Debugger, Device Driver, Compiler, Interpreter, Operating System (Basic Concepts only). SIC and SIC/XE Architecture, Addressing modes, SIC and SIC/XE Instruction set, Assembler Directives.

MODULE 2 (6 hours)

Assembly language programming and Assemblers:

SIC/XE Programming, Basic Functions of Assembler, Assembler Output Format – Header, Text and End Records. Assembler Data Structures, Two Pass Assembler Algorithm, Hand Assembly of SIC/XE Programs.

MODULE 3 (10 hours)

Assembler Features and Design Options:

Machine Dependent Assembler Features- Instruction Format and Addressing Modes, Program Relocation. Machine Independent Assembler Features –Literals, Symbol Defining Statements, Expressions, Program Blocks, Control Sections and Program Linking. Assembler Design Options- One Pass Assembler, Multi Pass Assembler. Implementation Example- MASM Assembler.

MODULE 4 (7 hours)

Loader and Linker:

Basic Loader Functions - Design of Absolute Loader, Simple Bootstrap Loader. Machine Dependent Loader Features- Relocation, Program Linking, Algorithm and Data Structures of Two Pass Linking Loader. Machine Independent Loader Features -Automatic Library Search, Loader Options. Loader Design Options.

MODULE 5 (8 hours)

Macro preprocessor, Device Driver, Text Editor and Debugger :

Macro Preprocessor - Macro Instruction Definition and Expansion, One pass Macro processor Algorithm and data structures, Machine Independent Macro Processor Features,

Macro processor design options. Device drivers - Anatomy of a device driver, Character and block device drivers, General design of device drivers. Text Editors- Overview of Editing, User Interface, Editor Structure. Debuggers - Debugging Functions and Capabilities, Relationship with other parts of the system, Debugging Methods- By Induction, Deduction and Backtracking.

Text Books

1. Leland L. Beck, System Software: An Introduction to Systems Programming, 3/E, Pearson Education Asia.

Reference Books

1. D.M. Dhamdhere, Systems Programming and Operating Systems, Second Revised Edition, Tata McGraw-Hill.
2. John J. Donovan, Systems Programming, Tata McGraw-Hill Edition 1991.
3. George Pajari, Writing UNIX Device Drivers, Addison Wesley Publications (Ebook : <http://tocs.ulb.tu-darmstadt.de/197262074.pdf>).
4. Peter Abel, IBM PC Assembly Language and Programming, Third Edition, Prentice Hall of India.
5. Jonathan Corbet, Alessandro Rubini, Greg Kroah-Hartman, Linux Device Drivers, Third Edition, O.Reilly Books
6. M. Beck, H. Bohme, M. Dziadzka, et al., Linux Kernel Internals, Second Edition, Addison Wesley Publications,
7. J Nithyashri, System Software, Second Edition, Tata McGraw-Hill.
8. The C Preprocessor <http://gcc.gnu.org/onlinedocs/gcc-2.95.3/cpp-1.html>

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to System Software		5 hours
1.1	System Software vs Application Software	1
1.2	Different System Software	1
1.3	SIC Architecture	1
1.4	SIC/XE Architecture	1

1.5	Assembler Directives	1
Module 2: Assembly language programming and Assemblers		6 hours
2.1	SIC Programming	1
2.2	SIC/XE Programming	1
2.3	Basic Functions of Assembler	1
2.4	Assembler Output Format and Data Structures	1
2.5	Two Pass Assembler Algorithm	1
2.6	Hand Assembly of SIC/XE Programs	1
Module 3: Assembler Features and Design Options		10 hours
3.1	Machine Dependent Assembler Features	1
3.2	Instruction Format and Addressing Modes	1
3.3	Program Relocation	1
3.4	Machine Independent Assembler Features	1
3.5	Literals, Symbol Defining Statements, Expressions	1
3.6	Program Blocks and Control Sections	1
3.7	Program Linking	1
3.8	Assembler Design Options- One Pass Assembler	1
3.9	Multi Pass Assembler	1
3.10	Implementation Example-MASM Assembler	1
Module 4: Loader and Linker		7 hours
4.1	Basic Loader Functions, Design of Absolute Loader, Simple Bootstrap Loader	1
4.2	Machine Dependent Loader Feature	1
4.3	Relocation, Program Linking	1
4.4	Algorithm and Data Structures of Two Pass Linking Loader	1
4.5	Machine Independent Loader Features	1
4.6	Automatic Library Search, Loader Options	1
4.7	Loader Design Options	1
Module 5: Macro preprocessor, Device Driver, Text Editor and Debugger		8 hours
5.1	Macro Preprocessor - Macro Instruction Definition and Expansion	1

5.2	One pass Macro processor Algorithm and data structures	1
5.3	Machine Independent Macro Processor Features, Macro processor design options	1
5.4	Device drivers - Anatomy of a device driver	1
5.5	Character and block device drivers, General design of device drivers	1
5.6	Text Editors- Overview of Editing, User Interface, Editor Structure	1
5.7	Debuggers -Functions and Capabilities, Relationship with other parts of the system	1
5.8	Debugging Methods- By Induction, Deduction and Backtracking	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. List out two system software and two application software.

Course Outcome 2 (CO 2):

1. How is upward compatibility between SIC and SIC/XE machines maintained?
2. Write a sequence of instructions for SIC/XE to divide BETA by GAMMA, setting ALPHA to the integer portion of the quotient and DELTA to the remainder. Use register- to-register instructions to make the calculation as efficient as possible.

Course Outcome 3 (CO 3):

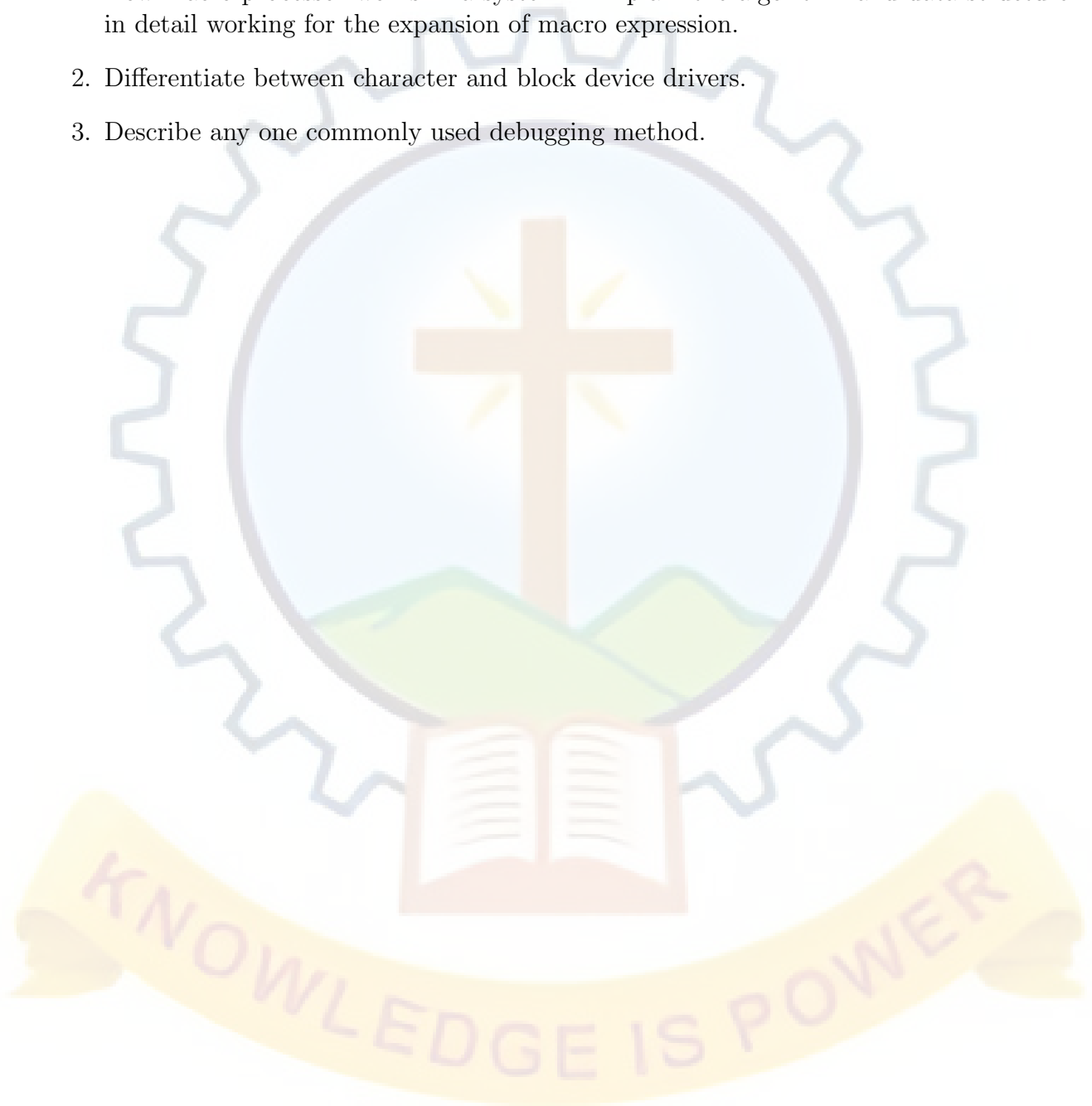
1. How do control sections and program blocks differ?
2. Can an assembler incorporating program blocks function using the same data structures as that of a normal two pass assembler? Justify your answer.
3. What are literals used for? Does the use of literals change the design of an assembler?
4. Design an assembler that can assemble a source program with different control sections.

Course Outcome 4 (CO 4):

1. How does the relocation and program linking works in a system ? Explain different loader design options.

Course Outcome 5 (CO 5):

1. How macro-processor works in a system ? Explain the algorithm and data structure in detail working for the expansion of macro expression.
2. Differentiate between character and block device drivers.
3. Describe any one commonly used debugging method.



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FOURTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2026

Course Code: B24CS2T07

Course Name: SYSTEM SOFTWARE

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Differentiate between system software and application software.
2. What are assembler directives? List out any five assembler directives in SIC.
3. Explain the different data structures used in the implementation of Assemblers.
4. List out the functions performed by an assembler.
5. What is a Literal? How is a literal handled by an assembler.
6. What are control sections? What is the advantage of using them?
7. Differentiate between linking loader and linkage editor? Which of these is preferable in a program development environment?
8. What is Automatic Library Search?
9. How should a programmer decide whether to use a macro or a subroutine to accomplish a given logical function?
10. Differentiate between character and block device drivers.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Differentiate between compilers and Interpreters. (4)
(b) Explain the architecture and addressing modes of SIC machine. (10)

OR

12. (a) Explain the addressing modes supported by SIC/ XE machine with suitable illustrations. (8)
(b) Explain the difference between i) A1 RESW 3 and A1 WORD 3 ii) B BYTE C'23' and B BYTE X'23' iii) END and END LABEL. (6)
13. (a) Let NUMBERS be an array of 100 words. Write a sequence of SIC/XE instructions to find the maximum of these numbers. (6)
(b) Perform hand assembly of the above written program using two pass assembler and show the status of various data structures and object program create. (8)

OR

14. (a) Write down and explain the second pass of a two pass assembler algorithm. (8)
(b) What is a Program Block. What is its advantage? With suitable example, explain how Program Blocks are handled by SIC assembler. (6)
15. (a) What is a Program Block. What is its advantage? With suitable example, explain how Program Blocks are handled by SIC assembler. (7)
(b) What is a forward reference? With example, illustrate how forward references are handled by a single pass assembler? (7)

OR

16. (a) With suitable examples explain machine dependent assembler features. (8)
(b) Explain with examples, the need and working of multipass assembler. (6)
17. (a) With the data structures used, state and explain two pass algorithm for a linking loader. (10)
(b) Explain about bootstrap loader. (4)

OR

18. (a) Explain about machine independent loader features. (9)
(b) What is Dynamic Linking? With example, illustrate how dynamic linking is performed. (5)
19. (a) Write down the single pass macro processor algorithm and with suitable example illustrate its working. (10)
(b) How are unique labels generated during Macro Expansion? (4)

OR

20. (a) Explain Text Editor structure in detail with a neat diagram. (7)
(b) Explain the different debugging methods in detail. (7)

B24CS2L05	LOGIC SYSTEM DESIGN AND OPERATING SYSTEMS LAB	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	3	2	2	2024

Preamble

This laboratory course integrates hands-on experimentation with digital logic circuits and core operating system concepts to strengthen practical understanding. In Part A, students will design and simulate combinational and sequential logic circuits—such as adders, counters, and shift registers—using open-source simulators, reinforcing foundational digital design principles. Part B focuses on operating system functionalities, including shell scripting, process management, CPU scheduling, synchronization, and memory management, implemented in Linux using C. Through these exercises, students will bridge theoretical knowledge with real-world applications, developing skills in problem-solving, system design, and performance evaluation. The course aligns with the syllabus of B24CS1T01 (Logic System Design) and B24CS2T04 (Operating Systems) for comprehensive assessment.

Prerequisites

Problem Solving and Programming Techniques(A)(B24ES1T01A), Data Structures(B24CS2T01), Logic System Design(B24CS1T01), Computer Organization and Architecture(B24CS2T02)

Course Outcomes

After the completion of the course the student will be able to

CO 1	Design and implement combinational logic circuits such as adders, multiplexers, demultiplexers, code converters, etc. and realize Boolean expressions using circuit simulators. (Cognitive Knowledge Level: Apply)
CO 2	Design various sequential circuits like shift registers, counters, etc. and simulate their working. (Cognitive Knowledge Level: Apply)
CO 3	Implement shell scripts using control structures and system commands. (Cognitive Knowledge Level: Apply)
CO 4	Apply system calls and IPC mechanisms to create, manage, and synchronize processes for concurrent execution. (Cognitive Knowledge Level: Apply)
CO 5	Evaluate CPU scheduling algorithms and page replacement strategies. (Cognitive Knowledge Level: Apply)
CO 6	Implement deadlock handling mechanisms to evaluate system state. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	2	2	2	1	1		1	1	2
CO 2	2	2	2	2	2	1	1		1	1	2
CO 3	2	2	2	2	2	1	1		1	1	2
CO 4	2	2	3	2	2	1	1		1	1	2
CO 5	2	2	3	2	2	1	1		1	1	2
CO 6	2	2	3	2	2	1	1		1	1	2

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Evaluation	20 marks
- Viva (Self Study)	10 marks
- Output	10 marks
Lab Record	5 marks
Viva-voce/Test	15 marks

SYLLABUS

LIST OF EXPERIMENTS

PART A	
1	Design and implement a combinational logic circuit for Half adder, full adder, Multiplexer, Demultiplexer, Encoder, Decoder.
2	Implement a Boolean function using MUX IC.
3	Design and implement the following shift registers using D flip flops (i) Serial in parallel out (ii) Parallel in parallel out.
4	Design and implement an asynchronous counter - N bit up counter, mod-N counter.
5	Design and implement a synchronous sequence generator.
PART B	
6	Shell programming: Write simple functions with condition and loop statements.
7	Implement system calls: fork, exec, getpid, wait, exit.
8	Implement inter-process Communication using shared memory.
9	Implement FCFS, SJF, Non-preemptive priority, Round-Robin CPU scheduling algorithms.
10	Solve one of the classic synchronization problems using semaphores.
11	Implement Banker's algorithm for deadlock avoidance.
12	Implement FIFO, OPT and LRU page replacement algorithms.

Reference Books

1. M. Morris Mano, Digital Logic Computer Design, 4/e, Pearson Education, 2013.
2. Thomas L Floyd, Digital Fundamentals, 10/e, Pearson Education, 2009.
3. M. Morris Mano, Computer System Architecture, 3/e, Pearson Education, 2007.
4. Andrew S Tanenbaum, "Modern Operating Systems" , 4th Edition, Prentice Hall, 2015.
5. William Stallings, "Operating systems", 6th Edition, Pearson, Global Edition, 2015.
6. Garry Nutt, Nabendu Chaki, Sarmistha Neogy, "Operating Systems", 3rd Edition, Pearson Education.

B24CS2L06	DATABASE LAB	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	3	2	2	2024

Preamble

This lab course provides hands-on experience in designing, implementing, and managing databases using SQL and NoSQL. Students will learn database schema design, querying, transaction control, and advanced features like stored procedures and triggers. The course bridges theory with practical application, preparing students for real-world database management.

Prerequisites

NIL

Course Outcomes

After the completion of the course the student will be able to

CO 1	Develop database schema for a given real-world problem domain using standard design and modeling approaches. (Cognitive Knowledge Level: Apply)
CO 2	Construct queries using SQL for database creation, interaction, modification, and updation. (Cognitive Knowledge Level: Apply)
CO 3	Plan and implement triggers and cursors, procedures, functions, and control structures using PL/SQL. (Cognitive Knowledge Level: Apply)
CO 4	Perform CRUD operations in NoSQL Databases. (Cognitive Knowledge Level: Apply)
CO 5	Design database applications using front-end tools and back-end DBMS. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	2								
CO 2	2	2	1		2						
CO 3		1	1		1						
CO 4	1	2	1	2	2						3
CO 5	1	1	1		2	2	1				2

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Evaluation	20 marks
- Viva (Self Study)	10 marks
- Output	10 marks
Lab Record	5 marks
Viva-voce/Test	15 marks

SYLLABUS

LIST OF EXPERIMENTS

1	Design a database schema for an application with ER diagram from a problem description.
2	Creation of database schema - DDL (create tables, set constraints, enforce relationships, create indices, delete and modify tables). Export ER diagram from the database and verify relationships (with the ER diagram designed in step 1).
3	Database initialization - Data insert, Data import to a database (bulk import using UI and SQL Commands).
4	Practice SQL commands for DML (insertion, updating, altering, deletion of data, and viewing/querying records based on condition in databases).
5	Implementation of various aggregate functions, Order By, Group By and Having clause in SQL.
6	Implementation of set operators nested queries, and join queries.
7	Practice of SQL TCL DCL commands like Rollback, Commit, Savepoint, Practice of SQL DCL commands for granting and revoking user privileges.

8	Practice of SQL commands for creation of views and assertions.
9	Creation and execution of stored procedures implementation of reusable function in PL/SQL.
10	Creation of event-driven actions in database using Triggers, Manage query results using explicit and implicit cursors.
11	Perform basic CRUD (Create, Read, Update, Delete) operations on a Cassandra table.
12	Perform CRUD operations with MongoDB.
13	Develop a database application using any front-end tool as a group project. The application constructed should have a minimum of five tables. Submit a detailed report of the project.

Reference Books

1. Sliberschatz Korth and S. Sudarshan, Database System Concepts, McGraw Hill, 7/e, 2017.
2. Adam Fowler, NoSQL for Dummies, John Wiley and Sons, 2015.
3. Olivier Pivert, NoSQL Data Models: Trends and Challenges (Computer Engineering: Databases and Big Data), Wiley 1/e, 2018.
4. Dan McCreary and Ann Kelly, Making the Sense of NoSQL: A guide for Managers and Rest of us, Manning 1/e, 2014.

B24CSM41	OBJECT ORIENTED PROGRAMMING IN JAVA(Minor)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

The purpose of this course is to enable learners to solve problems by breaking it down to object level while designing software and to implement it using Java. This course covers Object Oriented Principles, Object Oriented Programming in Java, Inheritance, Exception handling, Event handling, Multithreaded programming and working with window-based graphics. This course helps the learners to develop Mobile applications, Enterprise Applications, Scientific Applications and Web based Applications.

Prerequisites

Problem Solving and Programming Techniques(A) (B24ES1T01A)

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the fundamental concepts of object-oriented software design, the purpose and structure of UML diagrams, and the core components of the Java programming environment. (Cognitive Knowledge Level: Understand)
CO 2	Develop Java programs utilizing core language features such as primitive data types, operators, control statements, and object-oriented programming concepts including classes, objects, methods, and constructors. (Cognitive Knowledge Level: Apply)
CO 3	implement Java programs that demonstrate inheritance, polymorphism, packages, interfaces, and exception handling mechanisms to create robust and modular software.(Cognitive Knowledge Level: Apply)
CO 4	Utilize Java's advanced libraries for file handling, string processing, and collection management in practical applications. (Cognitive Knowledge Level: Apply)
CO 5	Construct interactive GUI applications using Swing components and event handling techniques and implement multithreaded programs for concurrent execution in Java.(Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	1	1			1		1				
CO 2	2		1	1							
CO 3	1		1	1		2					
CO 4	1			1	1						2
CO 5				1		2	2				2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (10 hours)

Introduction:

Approaches to Software Design - Functional Oriented Design, Object Oriented Design, Case Study of Automated Fire Alarm System.

Object Modeling Using UML – Basic Object Oriented concepts, UML (Unified Modeling Language) diagrams, Use case model, Class diagram, Interaction diagram, Activity diagram, State chart diagram.

Introduction to Java - Java programming Environment and Runtime Environment, Development Platforms -Standard, Enterprise. Java Virtual Machine (JVM), Java compiler, Bytecode, Java applet, Java Buzzwords, Java program structure, Comments, Garbage Collection, Lexical Issues.

MODULE 2 (9 hours)

Core Java Fundamentals:

Primitive Data types - Integers, Floating Point Types, Characters, Boolean. Literals, Type Conversion and Casting, Variables, Arrays, Strings, Vector class.

Operators - Arithmetic Operators, Bitwise Operators, Relational Operators, Boolean Logical Operators, Assignment Operator, Conditional (Ternary) Operator, Operator Precedence.

Control Statements - Selection Statements, Iteration Statements and Jump Statements.

Object Oriented Programming in Java - Class Fundamentals, Declaring Objects, Object Reference, Introduction to Methods, Constructors, this Keyword, Method Overloading, Using Objects as Parameters, Returning Objects, Recursion, Access Control, Static Members, Final Variables, Inner Classes, Command-Line Arguments, Variable Length Arguments.

MODULE 3 (10 hours)

More features of Java:

Inheritance - Super Class, Sub Class, The Keyword super, protected Members, Calling Order of Constructors, Method Overriding, the Object class, Abstract Classes and Methods, Using final with Inheritance.

Packages and Interfaces - Defining Package, CLASSPATH, Access Protection, Importing

Packages, Interfaces.

Exception Handling - Checked Exceptions, Unchecked Exceptions, try Block and catch Clause, Multiple catch Clauses, Nested try Statements, throw, throws and finally.

MODULE 4 (8 hours)

Advanced features of Java:

Input/Output - I/O Basics, Reading Console Input, Writing Console Output, PrintWriter Class, Object Streams and Serialization, Reading and Writing Files.

Java Library - String Handling – String Constructors, String Length, Special String Operations - Character Extraction, String Comparison, Searching Strings, Modifying Strings, Using valueOf(), Comparison of StringBuffer and String.

Collections framework – Collections overview, Collections Class – ArrayList. Accessing Collections via an Iterator.

MODULE 5 (8 hours)

GUI Programming, Event Handling and Multithreaded Programming:

Swing fundamentals - Swing Key Features, Model View Controller (MVC), Swing Controls, Components and Containers, Exploring Swing - JFrame, JLabel, JButton, JTextField.

Event handling - Event Handling Mechanisms, Delegation Event Model, Event Classes, Sources of Events, Event Listener Interfaces, Using the Delegation Model.

Multithreaded Programming - The Java Thread Model, The Main Thread, Creating Thread, Creating Multiple Threads, Suspending, Resuming and Stopping Threads.

Text Books

1. Herbert Schildt, Java: The Complete Reference, 8/e, Tata McGraw Hill, 2011.
2. Rajib Mall, Fundamentals of Software Engineering, 4th edition, PHI, 2014.
3. Paul Deitel, Harvey Deitel, Java How to Program, Early Objects 11th Edition, Pearson, 2018.

Reference Books

1. Y. Daniel Liang, Introduction to Java Programming, 7/e, Pearson, 2013.
2. Nageswararao R., Core Java: An Integrated Approach, Dreamtech Press, 2008.
3. Flanagan D., Java in A Nutshell, 8/e, O'Reilly, 2018.
4. Balagurusamy E., Programming JAVA a Primer, 5/e, McGraw Hill, 2014.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction		10 hours
1.1	Approaches to Software Design - Functional Oriented Design, Object Oriented Design	1
1.2	Case Study of Automated Fire Alarm System	1
1.3	Object Modeling Using UML – Basic Object Oriented concepts.	1
1.4	UML (Unified Modeling Language) diagrams, Use case model.	1
1.5	UML (Unified Modeling Language) diagrams, Class diagram, Interaction diagram.	1
1.6	UML (Unified Modeling Language) diagrams, Activity diagram, State chart diagram.	1
1.7	Introduction to Java - Java programming Environment and Runtime Environment.	1
1.8	Development Platforms -Standard, Enterprise. Java Virtual Machine (JVM)	1
1.9	Java compiler, Bytecode, Java applet, Java Buzzwords	1
1.10	Java program structure, Comments, Garbage Collection, Lexical Issues.	1
Module 2: Core Java Fundamentals		9 hours
2.1	Primitive Data types - Integers, Floating Point Types, Characters, Boolean. Literals, Type Conversion and Casting, Variables.	1
2.2	Arrays, Strings, Vector class.	1
2.3	Operators - Arithmetic Operators, Bitwise Operators, Relational Operators, Boolean Logical Operators, Assignment Operator, Conditional (Ternary) Operator, Operator Precedence.	1

2.4	Control Statements - Selection Statements, Iteration Statements and Jump Statements.	1
2.5	Object Oriented Programming in Java - Class Fundamentals, Declaring Objects, Object Reference	1
2.6	Introduction to Methods, Constructors, this Keyword.	1
2.7	Method Overloading, Using Objects as Parameters, Returning Objects	1
2.8	Recursion, Access Control, Static Members, Final Variables.	1
2.9	Inner Classes, Command-Line Arguments, Variable Length Arguments.	1
Module 3: More features of Java		10 hours
3.1	Inheritance - Super Class, Sub Class, The Keyword super,	1
3.2	Protected Members, Calling Order of Constructors	1
3.3	Method Overriding	1
3.4	The Object class, Abstract Classes and Methods, Using final with Inheritance.	1
3.5	Packages and Interfaces - Defining Package, CLASSPATH.	1
3.6	Access Protection, Importing Packages, Interfaces.	1
3.7	Exception Handling - Checked Exceptions, Unchecked Exceptions, try Block and catch Clause	1
3.8	Try Block and catch Clause	1
3.9	Multiple catch Clauses, Nested try Statements.	1
3.10	throw, throws and finally.	1
Module 4: Advanced features of Java		8 hours
4.1	TInput/Output - I/O Basics, Reading Console Input, Writing Console Output, PrintWriter Class	1
4.2	Object Streams and Serialization	1
4.3	Reading and Writing Files	1
4.4	Java Library - String Handling – String Constructors, String Length	1

4.5	Special String Operations - Character Extraction, String Comparison, Searching Strings, Modifying Strings	1
4.6	Using valueOf(), Comparison of StringBuffer and String	1
4.7	Collections framework – Collections overview, Collections Class – ArrayList. Accessing Collections via an Iterator	1
4.8	Accessing Collections via an Iterator	1
Module 5: GUI Programming, Event Handling, Multithreaded Programming		8 hours
5.1	Swing fundamentals - Swing Key Features, Model View Controller (MVC)	1
5.2	Swing Controls, Components and Containers	1
5.3	Exploring Swing - JFrame, JLabel, JButton, JTextField.	1
5.4	Event handling - Event Handling Mechanisms, Delegation Event Model.	1
5.5	Event Classes, Sources of Events.	1
5.6	Event Listener Interfaces, Using the Delegation Model.	1
5.7	Multithreaded Programming - The Java Thread Model, The Main Thread	1
5.8	Creating Thread, Creating Multiple Threads, Suspending, Resuming and Stopping Threads	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. For the following passage develop UML diagrams and then implement it as a Java program in accordance with your UML design.

Passage: College Office collects semester fee and college bus fee for each student. A clerk at the college office collects the fees from each student. The bus fee is calculated depending on the distance of the corresponding bus stop from the college. The semester fee varies depending upon the semester as well as branch of each student. Students are supposed to pay the fees in full. Economically backward students are eligible for 50% discount in semester fee. The consolidated fees receipt is issued to each student by the clerk, which contains the student name, admission number, semester and branch of student along with details of fees collected. Students can log in and view the details

of fees remitted and dues if any. The system allows students and clerk level login to the system. Clerk is able to view reports of each class showing status of fees payment of each student.

2. Draw a Class Diagram for an Online Shopping System including classes like User, Product, Order, Payment, and Cart. Show the relationships between them.

Course Outcome 2 (CO 2):

1. Write a Java program to prepare the rank list of students based on their performance in the Fourth Semester B.Tech. Degree examination. The output should be stored in a file.
2. Imagine a toll booth and a bridge. Cars passing by the booth are expected to pay an amount of Rs.50/- as toll tax. Mostly they do, but sometimes a car goes by without paying. The toll booth keeps track of the number of cars passed by, the total no. of the cars that has passed without paying and the total amount of money collected. Build a class 'Tollbooth' with two instance variables: a type unsigned int to hold the total number of cars and a type double to hold the total amount of money collected. A constructor should initialize both data members to zero. A method called payingCar() should increment the car total and add Rs.50/- to the cash total. Another method, called nopayCar(), increments the car total but adds nothing to the cash total. Finally, a method called display() displays the total no. of cars passed and the total amount collected.

Course Outcome 3 (CO 3):

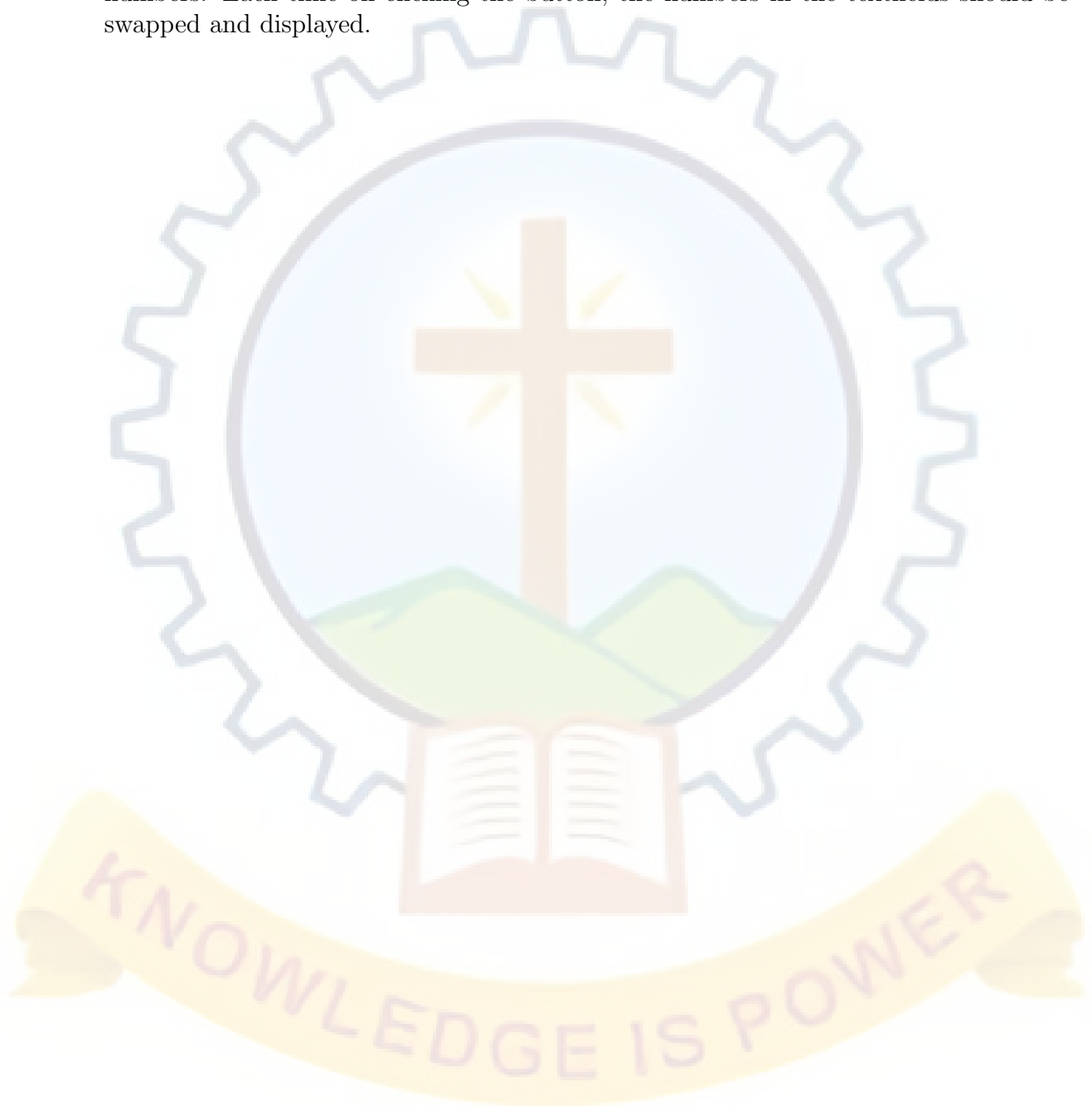
1. Write a program to demonstrate how event handling and exception handling are supported in Java.
2. Write sample programs to interpret the concepts as listed:
 - (a) Inheritance- Different types
 - (b) Abstract classes
 - (c) Method Overloading and Method Overriding

Course Outcome 4 (CO 4):

1. Write a program to demonstrate the start, run, sleep and join methods in Thread class.
2. Write a program that creates two threads- one to display the numbers from 1 to a given limit 'n' and the other thread to display 'n' letters of the English alphabet in order.

Course Outcome 5 (CO 5):

1. Write a GUI program to illustrate the use of JFrame, JTextField, JButton and JLabel by creating a login page
2. Write a GUI program that has two textfields and a button. Both textfields accept numbers. Each time on clicking the button, the numbers in the textfields should be swapped and displayed.



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FOURTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2026

Course Code: B24CSM41

***Course Name: OBJECT ORIENTED PROGRAMMING IN JAVA
(MINOR)***

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Briefly explain why Java is considered to be secure and portable.
2. Describe the concept of association among classes with an example.
3. Explain the different arithmetic operators in Java.
4. Explain the use for command line arguments with a suitable Java program.
5. Explain the use of CLASSPATH with an example.
6. What are the different types of exceptions?
7. Explain file handling features available in Java.
8. Write a simple program to read and print an integer value in Java.
9. Explain the concept of main thread in multi-threading.
10. Explain any two Event classes in Java.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Describe in detail polymorphism, abstraction and inheritance with suitable examples. (9)
(b) Explain the role of Java Virtual Machine. (5)

OR

12. (a) Compare and contrast Functional Oriented and Object Oriented approach by considering a simple bus ticket reservation system. (5)
(b) Explain class diagram with an example. (9)
13. (a) Explain primitive data types in Java. How are they different from other data types? (8)
(b) Explain variables and arrays in Java. (6)

OR

14. (a) Using a suitable program, explain the concept of methods and constructors. (8)
(b) Explain the keyword super and its usage in Java. (6)
15. (a) Using a table, explain the effect of access specifiers in inheritance. (6)
(b) Describe in detail about exception handling using try block and catch clause in Java with the help of a suitable Java program. (8)

OR

16. (a) What is an interface in Java? Explain with a suitable example. (8)
(b) Explain throw, throws and finally constructs with the help of a Java program. (6)
17. (a) Explain ArrayList collections framework. Also explain the use of iterator in accessing collections. (8)
(b) Bring out the difference between “==” and equals() method with the help of a sample program. (6)

OR

18. (a) Compare Byte Streams and Character Streams. Write a program to demonstrate the usage of the PrintWriter class. (8)
(b) Explain any three String constructors with the help of sample code for each. (6)
19. (a) Explain in detail the Delegation Event model for event handling in Java. (7)
(b) Describe in detail the creation of a thread using the Runnable interface. (7)

OR

20. (a) What are the differences between a process and a thread? (4)
(b) Write a Graphical User Interface (GUI) based Java program to implement a simple calculator supporting the operations addition, subtraction, multiplication and division. Use Swing controls to implement GUI. There may be three text boxes, the first two for operands and the last for result. Add four buttons for the above operations. Write neat comments in your program to show how you handle events. (10)

B24CSM42	INTRODUCTION TO MACHINE LEARNING (Minor)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		2024

Preamble

This course enables the learners to understand the basic concepts and algorithms in machine learning. The course covers the standard and most popular supervised learning algorithms such as linear regression, logistic regression, decision trees, Naive Bayes algorithm, basic clustering algorithms, and classifier performance measures. This course helps students to provide machine learning-based solutions to real-world problems.

Prerequisites

Basic understanding of probability theory and linear algebra.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Understand machine learning paradigms and apply parameter estimation techniques for model optimization. (Cognitive Knowledge Level: Understand)
CO 2	Understand and implement neural networks with backpropagation, analyze training challenges such as vanishing gradients and overfitting, and apply adaptive learning rate methods like ADAGRAD, RMSProp, and ADAM. (Cognitive Knowledge Level: Apply)
CO 3	Apply supervised learning algorithms like Naive Bayes, KNN, SVM, and Decision Trees; perform model training and validation; and evaluate regression models using MAE, RMSE, and R-squared metrics. (Cognitive Knowledge Level: Apply)
CO 4	Apply clustering techniques and similarity measures, perform dimensionality reduction using PCA and MDS, and implement feature selection methods like entropy, correlation, and chi-square tests. (Cognitive Knowledge Level: Apply)
CO 5	Apply ensemble learning techniques such as voting, bagging, boosting, and stacking using models like Random Forest, and assess classification model performance using metrics including confusion matrix, precision, recall, F1-score, ROC, and AUC. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	2	1	2	3						2
CO 2	3	2	2	2	3						2
CO 3	3	3	3	3	3						2
CO 4	3	2	2	3	3						2
CO 5	3	3	2	3	3						2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Introduction to ML:

Machine Learning vs. Traditional Programming, Machine learning paradigms - supervised, semi-supervised, unsupervised, reinforcement learning. Types of ML problems: Classification, Clustering and Regression. Basics of parameter estimation - maximum likelihood estimation (MLE) and maximum a posteriori estimation (MAP), Bayesian formulation. Supervised Learning. Feature Representation and Problem Formulation, Role of loss functions and optimization. Regression - Linear regression with one variable, Linear regression with multiple variables - solution using gradient descent algorithm and matrix method, Non-linear Regression.

MODULE 2 (10 hours)

Neural Networks (NN):

Perceptron, Neural Network - Multilayer feed-forward network, Activation functions (Sigmoid, ReLU, Tanh), Back propagation algorithm, Issues in Back Propagation Learning (Vanishing and Exploding Gradient Problems, Local Minima and Saddle Points, Overfitting, Slow Convergence, Sensitivity to the selection of the Initial weights, Computational Cost), Use of Adaptive Learning Rate Methods (Back Propagation Algorithm with Momentum, Adaptive Gradient Descent Algorithm (ADAGRAD), Root Mean Square Propagation (RMSProp), Adaptive Momentum Estimation (ADAM)).

MODULE 3 (10 hours)

Classification Methods:

Naive Bayes Classifier - Working Principle, Types of Naive Bayes Classifiers, Illustrative examples. K-Nearest Neighbour Classifier (KNN):- Working Principle, Illustrative examples on designing KNN Classifiers; Support Vector machines (SVM) - Maximum Margin Classification. Decision Trees – Information Gain, Gain Ratio, ID3 algorithm. Generalisation and Overfitting - Idea of overfitting, LASSO and RIDGE regularization, Idea of Training, Testing, Validation. Regression - Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R Squared/Coefficient of Determination.

MODULE 4 (8 hours)

Clustering Methods:

Clustering - Similarity measures, Hierarchical Clustering - Agglomerative Clustering, partitional clustering, Expectation maximization (EM) for soft clustering, K-means clustering, Density Based clustering. Feature selection techniques: Entropy, Correlation Coefficient, Chi-square Test, Forward & Backward Selection; Dimensionality reduction: Principal Component Analysis, Multidimensional scaling.

MODULE 5 (8 hours)

Ensemble Learning:

Basic Techniques (Voting, Simple and Weighted Averaging Techniques), Advanced Techniques (Bagging, Boosting, Stacking) - The Random Forest Classifier. Assessment of Classification Models: Performance measures- Confusion Matrix, Accuracy, Precision, Recall, F1-Score, Receiver Operating Characteristics Curve (ROC), Area Under the Curve (AUC), Illustrative examples on Classification model assessment.

Text Books

1. Tom Mitchell, Machine Learning, McGraw-Hill, 1997.
2. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements Of Statistical Learning, Second edition Springer 2007.
3. Shai Shalev-Shwartz, Shai Ben-David, Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press, 2014.
4. Ethem Alpaydin, Introduction to Machine Learning, 2nd edition, MIT Press 2010.

Reference Books

1. M Gopal, Applied Machine Learning, McGraw Hill 2/e, 2018.
2. P. Langley, Elements of Machine Learning, Morgan Kaufmann, 1995.
3. Manaranjan Pradhan, U Dinesh Kumar, Machine Learning using Python, Wiley, 1/e, 2019.
4. M.N. Murty, V.S. Ananthanarayana, Machine Learning: Theory and Practice, Universities Press 1/e, 2024.
5. Richert and Coelho, Building Machine Learning Systems with Python.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction to ML		9 hours
1.1	Machine Learning vs. Traditional Programming	1
1.2	Machine learning paradigms - supervised, semi-supervised, unsupervised, reinforcement learning.	1
1.3	Types of ML problems: Classification, Clustering and Regression.	1
1.4	Basics of parameter estimation – maximum likelihood estimation (MLE) and maximum a posteriori estimation (MAP), Bayesian formulation.	1
1.5	Supervised Learning - Feature Representation and Problem Formulation, Role of loss functions and optimization.	1
1.6	Regression - Linear regression with one variable	1
1.7	Linear regression with multiple variables	1
1.8	solution using gradient descent algorithm and matrix method	1
1.9	Non-linear Regression	1
Module 2: Neural Networks (NN)		10 hours
2.1	Perceptron, Neural Network	1
2.2	Multilayer feed-forward network	1
2.3	Activation functions(Sigmoid, ReLU, Tanh)	1
2.4	Back propagation algorithm	1
2.5	Issues in Back Propagation Learning (Vanishing and Exploding Gradient Problems	1
2.6	Local Minima and Saddle Points, Overfitting	1
2.7	Slow Convergence, Sensitivity to the selection of the Initial weights, Computational Cost)	1

2.8	Use of Adaptive Learning Rate Methods (Back Propagation Algorithm with Momentum	1
2.9	Adaptive Gradient Descent Algorithm (ADAGRAD), Root Mean Square Propagation (RMSProp)	1
2.10	1. Adaptive Momentum Estimation (ADAM)).	1
Module 3: Classification Methods		10 hours
3.1	Naive Bayes Classifier - Working Principle	1
3.2	Types of Naive Bayes Classifiers, Illustrative examples.	1
3.3	K-Nearest Neighbour Classifier (KNN): Working Principle, Illustrative examples on designing KNN Classifiers	1
3.4	Support Vector machines (SVM)	1
3.5	Maximum Margin Classification	1
3.6	Decision Trees – Information Gain, Gain Ratio, ID3 algorithm	1
3.7	Generalisation and Overfitting - Idea of overfitting	1
3.8	LASSO and RIDGE regularization	1
3.9	Idea of Training, Testing, Validation	1
3.10	Regression - Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R Squared/Coefficient of Determination	1
Module 4: Clustering Methods		8 hours
4.1	Clustering - Similarity measures	1
4.2	Hierarchical Clustering - Agglomerative Clustering, partitional clustering	1
4.3	Expectation maximization (EM) for soft clustering	1
4.4	K-means clustering, Density Based clustering	1
4.5	Feature selection techniques: Entropy, Correlation Coefficient	1
4.6	Chi-square Test, Forward & Backward Selection	1
4.7	Dimensionality reduction: Principal Component Analysis	1
4.8	Multidimensional scaling	1

Module 5: Ensemble Learning		8 hours
5.1	Ensemble Learning: Basic Techniques (Voting, Simple and Weighted Averaging Techniques)	1
5.2	Advanced Techniques (Bagging, Boosting, Stacking)	1
5.3	The Random Forest Classifier	1
5.4	Assessment of Classification Models- Performance measures	1
5.5	Confusion Matrix, Accuracy, Precision, Recall, F1-Score	1
5.6	Receiver Operating Characteristics Curve (ROC)	1
5.7	Area Under the Curve (AUC)	1
5.8	Illustrative examples on Classification model assessment	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. A dataset is provided where the output variable is known for some data points and unknown for others. Propose an appropriate machine learning paradigm to handle this and justify your choice.
2. Compare and contrast MLE, MAP, and MMSE in terms of their assumptions and optimality criteria. In what scenarios would one method be preferred over the others?

Course Outcome 2 (CO 2):

1. Implement a simple two-layer feedforward neural network with backpropagation on a binary classification problem.
2. Compare ADAGRAD, RMSProp, and ADAM optimizers. In which scenario would you prefer ADAM over the others?

Course Outcome 3 (CO 3):

1. Demonstrate how soft-margin SVM handles misclassification. Provide mathematical constraints and an illustrative example.

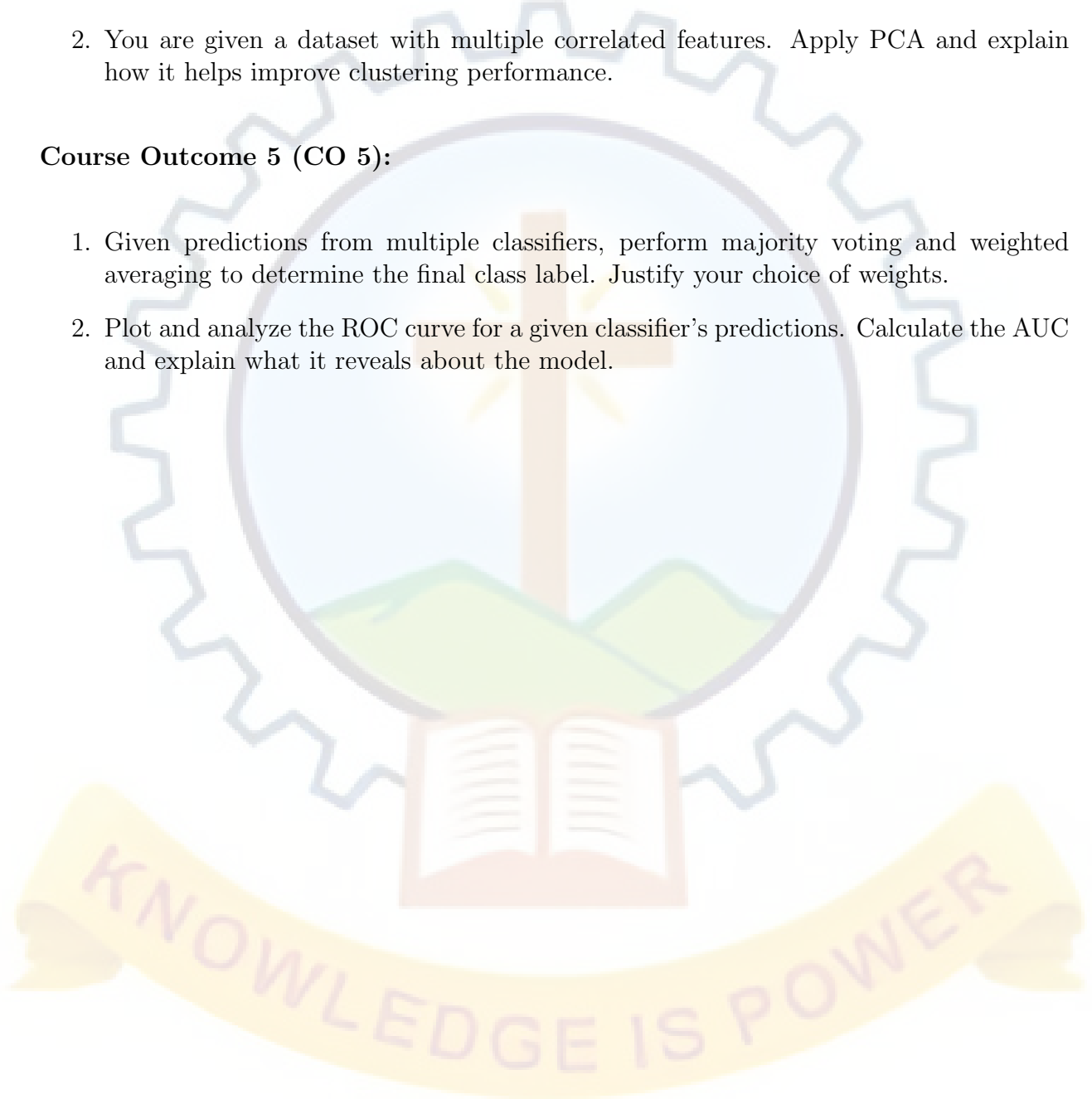
2. Using a provided dataset, build a Naïve Bayes classifier and compute the posterior probabilities for classification. Indicate how prior and likelihood are used.

Course Outcome 4 (CO 4):

1. Perform K-means clustering on a sample dataset with $k = 2$. Show initial centroid selection, assignment, and update steps.
2. You are given a dataset with multiple correlated features. Apply PCA and explain how it helps improve clustering performance.

Course Outcome 5 (CO 5):

1. Given predictions from multiple classifiers, perform majority voting and weighted averaging to determine the final class label. Justify your choice of weights.
2. Plot and analyze the ROC curve for a given classifier's predictions. Calculate the AUC and explain what it reveals about the model.



MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FOURTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2026

Course Code: B24CSM42

Course Name: INTRODUCTION TO MACHINE LEARNING (MINOR)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Differentiate between supervised, unsupervised, and semi-supervised learning with suitable examples.
2. What is the role of parameter estimation in machine learning models?
3. Explain why logistic regression is used for classification instead of regression.
4. What is the bias-variance trade-off? How does it affect model selection?
5. Differentiate between hard margin SVM and soft margin SVM.
6. Why is KNN considered a lazy learning algorithm?
7. What is a similarity measure in clustering? Name any two commonly used measures.
8. Mention two differences between Hierarchical Clustering and K-means Clustering.
9. Differentiate between bagging and boosting.
10. Explain the concept of weighted averaging in ensemble models.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain how reinforcement learning can be used to train autonomous agents, such as self-driving cars. Discuss key challenges in implementing RL-based systems. (7)
- (b) Describe a real-world problem that can be solved using unsupervised learning. Discuss the steps involved in implementing a machine learning solution for this problem. (7)

OR

12. (a) Compare MLE, MAP, and MMSE in terms of their assumptions, computational complexity, and suitability for different types of data. (7)
- (b) Explain the significance of regularization in parameter estimation. How do L1 and L2 regularization techniques affect MLE-based models? (7)
13. (a) Compare and contrast the activation functions: Sigmoid, Tanh, and ReLU. Discuss their impact on the vanishing gradient problem in deep networks. (7)
- (b) Discuss in detail the challenges faced during the training of deep neural networks using backpropagation, such as vanishing/exploding gradients, overfitting, local minima, and computational cost. Provide strategies to overcome them. (7)

OR

14. (a) Explain the concept of adaptive learning rate methods in training neural networks. Compare Backpropagation with Momentum, ADAGRAD, RMSProp, and ADAM in terms of convergence and performance. (7)
- (b) Demonstrate with examples how momentum and adaptive optimizers improve gradient descent. Discuss how these methods impact convergence rate and stability during neural network training. (7)
15. (a) Compare hard margin SVM and soft margin SVM. Explain how the introduction of slack variables helps in handling non-linearly separable data. (7)
- (b) Explain different types of kernel functions in SVM. Discuss how choosing an appropriate kernel affects model performance. (7)

OR

16. (a) Compare and contrast KNN with SVM in terms of computational complexity, decision boundaries, and real-world applicability. (7)
- (b) Discuss the advantages and disadvantages of KNN. How does KNN perform on high-dimensional data? Suggest techniques to overcome challenges. (7)
17. (a) Explain how eigenvectors and eigenvalues are used in PCA to determine the principal components. Provide a numerical illustration if possible. (7)

- (b) Describe the Hierarchical Agglomerative Clustering method. How does the dendrogram help in choosing the number of clusters? (7)

OR

18. (a) Given a dataset with high-dimensional features, explain how PCA reduces dimensionality and improves performance. Discuss its limitations. (7)
- (b) Explain how eigenvectors and eigenvalues are used in PCA to determine the principal components. Provide a numerical illustration if possible. (7)
19. (a) Describe in detail the working of boosting algorithms. How does it focus on difficult-to-classify instances? (7)
- (b) Explain voting and averaging techniques in ensemble learning. Provide examples where each is suitable. (7)

OR

20. (a) Explain how ROC and AUC are used to evaluate classification models. Draw a sample ROC curve and interpret its meaning. (7)
- (b) Given a classification model with the following confusion matrix, calculate accuracy, precision, recall, and F1-score. Discuss what these values indicate about the model's performance. . (7)

B24CSM43	INTRODUCTION TO COMPUTER NETWORKS (Minor)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course introduces students a clear understanding of how computer networks from local area networks to the massive and global Internet are built, how they allow computers to share information and communicate with one another. This course explores the physical aspects of computer networks, the layers of the OSI Reference Model, and inter-networking. It enables learners to compare and analyze existing network technologies, helping them select the most suitable network design for a given system.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Comprehend the OSI and TCP/IP models, the functioning of different network layers, and the protocol stack used in computer networks. (Cognitive Knowledge Level: Understand)
CO 2	Analyze data link layer functionalities, error detection and correction techniques, and medium access control protocols for efficient data transmission. (Cognitive Knowledge Level: Apply)
CO 3	Demonstrate an understanding of network layer operations, including addressing, routing algorithms, and congestion control mechanisms. (Cognitive Knowledge Level: Apply)
CO 4	Examine transport layer services, protocols, congestion control mechanisms. (Cognitive Knowledge Level: Apply)
CO 5	Understand client-server and peer-to-peer applications like HTTP, FTP, DNS, and P2P networks. (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	2	2	1				3		3
CO 2	2	3	3	3	3	1					3
CO 3	3	3	2	3	3	2					3
CO 4	2	3	3	3	3	2					3
CO 5	2	2	2	2	2	3					3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (10 hours)

Introduction & Physical Layer

Introduction, Uses of computer networks, Network Hardware, Local Area Networks (LAN), Metropolitan Area Networks (MAN), Wide Area Networks (WAN), Wireless networks, Home networks, Internetworks, Network Software, Protocol hierarchies, Design issues for the layers, Connection-oriented and Connectionless services, Service primitives, Relationship of services to protocols, Reference models, The OSI reference model, The TCP/IP reference model, Comparison of OSI and TCP/IP reference models.

Physical layer- Modes of communication, Physical topologies, Signal encoding, Transmission media- Guided media, Unguided/wireless media, Performance indicators, Bandwidth, Throughput, Latency, Queuing time, Bandwidth-Delay product.

MODULE 2 (11 hours)

Data Link Layer

Data Link Layer: Functions and Services, Error detection and correction, Error detecting codes, Error correcting codes. Flow Control Mechanisms: Stop-and-Wait, Sliding Window Protocols, High-Level Data Link Control(HDLC)protocol. Medium Access Control (MAC) Protocols: ALOHA, CSMA/CD, CSMA/CA, IEEE 802.3 Ethernet (Wired LAN), Ethernet performance, Switched Ethernet, Fast Ethernet, Gigabit Ethernet, IEEE 802.2: Logical Link Control, IEEE 802.11 (Wireless LAN), 802.11 protocol stack, Physical layer, MAC Sublayer protocol, Frame structure, Mobile IP. Bridges and switches- Bridges from 802.x to 802.y, Repeaters, Hubs, Bridges, Switches, Routers, and Gateways.

MODULE 3 (9 hours)

Network Layer

Introduction: Network Layer Services & Functions, Logical addressing- IPv4 and IPv6 addresses, Network-layer protocols - Internet Protocol- IPV4 and IPv6, Unicast routing protocols- Distance Vector Routing, Link State Routing, Multicast routing - Multicasting Basics, Intra domain and inter-domain routing, Next generation IP, Congestion Control Techniques (Leaky Bucket, Token Bucket), Quality of Service (QoS) Techniques for Network Performance.

MODULE 4 (7 hours)

Transport Layer

Transport Layer Protocols: User Datagram Protocol (UDP)– Features and Applications, Transmission Control Protocol (TCP)– Features, Segment Structure, TCP Connection Establishment and release, TCP retransmission policy, Congestion Control- Open Loop Vs Closed Loop Congestion Control, Congestion Control in TCP.

MODULE 5 (8 hours)

Application Layer and Network Security

Application Layer: Application-Layer Protocols, Client-server applications - World Wide Web and HTTP, FTP, Electronic Mail, DNS, Peer-to-peer paradigm - P2P Networks.
Network Security Basics: Symmetric/asymmetric encryption, Digital Signatures, SSL/TLS, Common Attacks: DoS, Spoofing, Sniffing.

Text Books

1. Behrouz A Forouzan, Data Communication and Networking, 4/e, Tata McGraw Hill
2. Andrew S. Tanenbaum, Computer Networks, 4/e, PHI (Prentice Hall India).

Reference Books

1. Larry L Peterson and Bruce S Dave, Computer Networks – A Systems Approach, 5/e, Morgan Kaufmann
2. Fred Halsall, Computer Networking and the Internet, 5/e.
3. James F. Kurose, Keith W. Ross, Computer Networking: A Top-Down Approach, 6/e.
4. Keshav, An Engineering Approach to Computer Networks, Addison Wesley, 1998.
5. W. Richard Stevens. TCP/IP Illustrated Volume 1, Addison-Wesley, 2005.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction & Physical Layer		10 hours
1.1	Introduction, Uses of computer networks	1
1.2	Network Hardware, Local Area Networks (LAN), Metropolitan Area Networks (MAN), Wide Area Networks (WAN), Wireless networks, Home networks, Internetworks	1
1.3	Network Software, Protocol hierarchies, Design issues for the layers	1
1.4	Connection-oriented and Connectionless services, Service primitives, Relationship of services to protocols	1
1.5	Reference models, The OSI reference model	1
1.6	The TCP/IP reference model, Comparison of OSI and TCP/IP reference models	1
1.7	Physical layer- Modes of communication, Physical topologies	1
1.8	Signal encoding	1
1.9	Transmission media- Guided media, Unguided/wireless media	1
1.10	Performance indicators, Bandwidth, Throughput, Latency, Queuing time, Bandwidth-Delay product	1
Module 2: Data Link Layer		11 hours
2.1	Data Link Layer: Functions and Services	1
2.2	Error detection and correction, Error detecting codes	1
2.3	Error correcting codes	1
2.4	Flow Control Mechanisms: Stop-and-Wait, Sliding Window Protocols	1
2.5	High-Level Data Link Control(HDLC)protocol	1
2.6	Medium Access Control (MAC) Protocols: ALOHA, CSMA/CD, CSMA/CA	1

2.7	IEEE 802.3 Ethernet (Wired LAN)	1
2.8	Ethernet performance, Switched Ethernet, Fast Ethernet, Gigabit Ethernet, IEEE 802.2: Logical Link Control	1
2.9	IEEE 802.11 (Wireless LAN), 802.11 protocol stack, Physical layer, MAC Sublayer protocol, Frame structure	1
2.10	Mobile IP	1
2.11	Bridges and switches, Bridges from 802.x to 802.y, Repeaters, Hubs, Bridges, Switches, Routers, and Gateways	1
Module 3: Network Layer		9 hours
3.1	Introduction : Network Layer Services & Functions	1
3.2	Logical addressing- IPv4 and IPv6 addresses	1
3.3	Network-layer protocols - Internet Protocol- IPV4 and IPv6	1
3.4	Unicast routing protocols- Distance Vector Routing	1
3.5	Link State Routing	1
3.6	Multicast routing - Multicasting Basics	1
3.7	Intra domain and inter-domain routing, Next generation IP	1
3.8	Congestion Control Techniques (Leaky Bucket, Token Bucket)	1
3.9	Quality of Service (QoS) Techniques for Network Performance	1
Module 4: Transport Layer		7 hours
4.1	Transport Layer Services & Protocols	1
4.2	User Datagram Protocol (UDP) – Features and Applications	1
4.3	Transmission Control Protocol (TCP) – Features, Segment Structure	1
4.4	TCP Connection Establishment and release	1
4.5	TCP retransmission policy	1
4.6	Open Loop Vs Closed Loop Congestion Control	1

4.7	TCP Congestion Control Mechanisms	1
Module 5: Application Layer and Network Security		8 hours
5.1	Overview of Application Layer Protocols, Client-server applications, Domain Name System (DNS) and its Role in the Internet	1
5.2	Hypertext Transfer Protocol (HTTP) and World Wide Web (WWW)	1
5.3	File Transfer Protocol (FTP) and Simple Mail Transfer Protocol (SMTP)	1
5.4	Peer-to-peer paradigm - P2P Networks	1
5.5	Network Security Basics: Symmetric/asymmetric encryption	1
5.6	Digital Signatures	1
5.7	SSL/TLS	1
5.8	Common Attacks: DoS, Spoofing, Sniffing.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Compare TCP/IP and OSI reference model.
2. The purpose of physical layer is to transport a raw bit stream from one machine to another. Justify.

Course Outcome 2 (CO 2):

1. Ethernet frames must be at least 64 bytes long to ensure that the transmitter is still going in the event of a collision at the far end of the cable. Fast Ethernet has the same 64-byte minimum frame size but can get the bits out ten times faster. How is it possible to maintain the same minimum frame size?
2. What do you mean by bit stuffing?

Course Outcome 3 (CO 3):

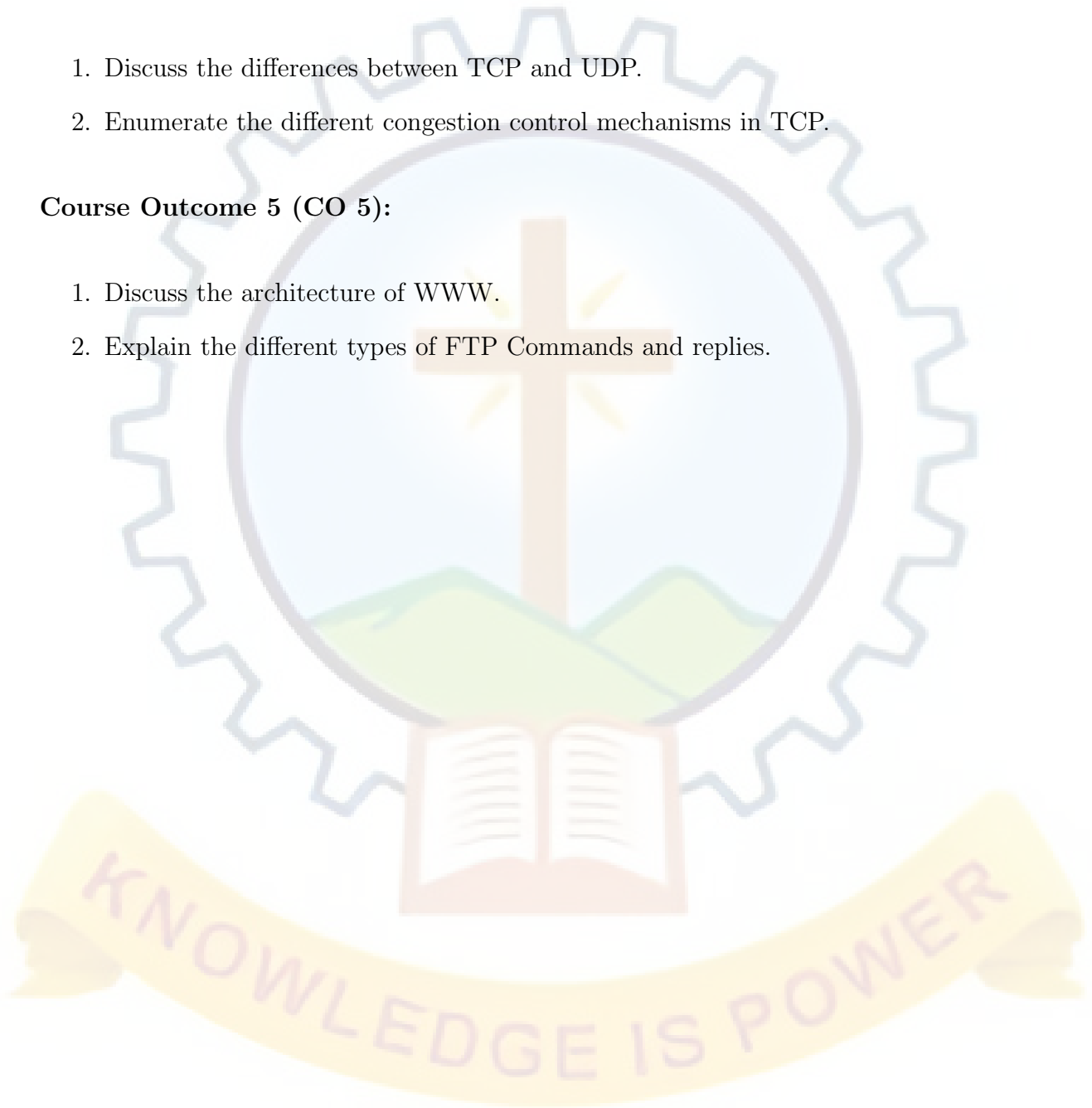
1. Discuss the differences between IPV4 and IPV6.
2. Explain the different types of routing algorithms.

Course Outcome 4 (CO 4):

1. Discuss the differences between TCP and UDP.
2. Enumerate the different congestion control mechanisms in TCP.

Course Outcome 5 (CO 5):

1. Discuss the architecture of WWW.
2. Explain the different types of FTP Commands and replies.



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FOURTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2026

Course Code: B24CSM43

***Course Name: INTRODUCTION TO COMPUTER NETWORKS
(MINOR)***

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. What does “negotiation” mean when discussing network protocols in a layered architecture? Give an example.
2. Define simplex, half-duplex, and full-duplex transmission modes. Give one example for each.
3. Data link protocols almost always put the CRC in a trailer rather than in a header. Why?
4. An 8-bit byte with binary value 10101111 is to be encoded using an even-parity Hamming code. What is the binary value after encoding?
5. Illustrate the Count to Infinity problem in routing.
6. Discuss the IPV6 frame format.
7. Explain the TCP Segment Header.
8. Discuss the TCP Connection Establishment method.
9. Explain the role of DNS in Internet.

10. What is SMTP. Explain its features.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Discuss the ISO-OSI model with the help of a neat diagram. (14)

OR

12. (a) Explain the different signal encoding techniques in the physical layer (8)
(b) Compare the TCP/IP layer with the ISO-OSI layer in the physical layer. (6)
13. (a) Discuss the GoBackN and selective repeat sliding window protocols. (8)
(b) Differentiate IEEE 802.3 with IEEE 802.11 (6)

OR

14. (a) Differentiate the different Medium Access Control(MAC) protocols. (14)
15. (a) Explain the different types of routing algorithms. (14)

OR

16. (a) What are the techniques for ensuring good QoS? (8)
(b) Discuss the roles of intra-domain and inter-domain routing protocols with examples. (6)
17. (a) Discuss the structure of TCP Header. (8)
(b) What are the TCP Congestion control mechanisms. (6)

OR

18. (a) What are the differences between TCP and UDP (8)
(b) Describe the structure of UDP Header. (6)
19. (a) Discuss DNS and its role in Internet? (8)
(b) Explain the different types of FTP Queries and Replies? (6)

OR

20. (a) Discuss the features of SMTP? (8)
(b) Explain the features of WWW? (6)

B24CSH41	ADVANCED TOPICS IN COMPUTER GRAPHICS (Honours)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		2024

Preamble

This course helps the learners to make awareness about strong theoretical concept in computer graphics. It covers the three-dimensional environment representation in a computer, transformation of 2D/3D objects and basic mathematical techniques and algorithms used to build applications. This course enables the learners to develop the ability to create image processing frameworks for different domains and develop algorithms for emerging display technologies.

Prerequisites

A sound knowledge of Mathematics and concepts of any programming language.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Describe the working principles of graphics devices and multimedia concepts. (Cognitive Knowledge level: Understand)
CO 2	Illustrate line drawing, circle drawing and polygon filling algorithms. (Cognitive Knowledge level: Apply)
CO 3	Demonstrate geometric representations and transformations on 2D and 3D objects. (Cognitive Knowledge level: Apply)
CO 4	Demonstrate the working of line and polygon clipping algorithms. (Cognitive Knowledge level: Apply)
CO 5	Summarize visible surface detection methods and illustrate projection algorithms. (Cognitive Knowledge level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	3			2	1			3		2
CO 2	2	3	2	3	3						2
CO 3	2	3	2	3	3						2
CO 4	2	3	2	3	3						2
CO 5	2	3	2	3	3						2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (10 hours)

Basics of Computer graphics and Fundamental of Multimedia:

Basics of Computer Graphics and its applications. Video Display devices- Refresh Cathode Ray Tubes(CRT), Random Scan Displays and systems, Raster scan displays and systems, Video Display devices- LED, OLED, Reflective displays and Touch Panel Technology.

Introduction to Multimedia, Authoring and Tools, Graphics and Image Data Representations, Popular File Formats, Fundamental Concepts and types of Video, Basics of Digital Audio and its types.

MODULE 2 (8 hours)

Line and Shape Drawing Algorithms:

Line drawing algorithms- DDA, Bresenham's algorithm. Circle drawing algorithms- Midpoint Circle generation algorithm, Bresenham's algorithm. Ellipse-Generating Algorithms, Properties of Ellipses, Midpoint Ellipse Algorithm.

Filled Area Primitives- Scan line polygon filling, Boundary filling and flood filling.

MODULE 3 (10 hours)

Geometric transformations:

Two-dimensional transformations- Translation, Rotation, Scaling, Reflection and Shearing, Composite transformations- General Pivot-Point Rotation, General Fixed-Point Scaling, General Scaling Directions, Matrix representations and homogeneous coordinates. Basic 3D transformations.

MODULE 4 (9 hours)

Transformations and Clipping Algorithms :

Window to viewport transformation, Cohen Sutherland and Midpoint subdivision line clipping algorithms, Liang-Barsky Line Clipping, Nicholl-Lee-Nicholl Line Clipping, Sutherland Hodgeman and Weiler Atherton Polygon clipping algorithms, Text Clipping, Exterior Clipping

MODULE 5 (8 hours)

Three dimensional graphics and Graphics Functions:

Three dimensional viewing pipeline. Projections- Parallel and Perspective projections.

Visible surface detection algorithms- Back face detection, Depth buffer algorithm, Scan line algorithm, A-Buffer Method, Ray-Casting Method. C Graphics Basics Graphics programming, initializing the graphics, C Graphical functions, simple programs.

Text Books

1. Zhigang Xiang and Roy Plastock, Computer Graphics (Schaum's outline Series), McGraw Hill, 2019.
2. Donald Hearn and M. Pauline Baker, Computer Graphics, Pearson Education, 2/e 2002.
3. Ze-Nian Li and Mark S. Drew, Fundamentals of Multimedia, Pearson Education, 3/e 2021.

Reference Books

1. D. P. Mukherjee, Debasish Jana, Computer Graphics : Algorithms and Implementations, Prentice Hall India, 2/e, 2010.
2. Zhigang Xiang and Roy Plastock, Computer Graphics, McGraw Hill, 2/e, 2020.
3. David F. Rogers, Procedural Elements for Computer Graphics, McGraw Hill, 2/e, 2017.
4. Zhigang Xiang and Roy Plastock, Computer Graphics, McGraw Hill, 2/e, 2020.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Basics of Computer graphics and Fundamental of Multimedia		10 hours
1.1	Basics of Computer Graphics and applications	1
1.2	Refresh Cathode Ray Tubes	1
1.3	Random Scan Displays and systems	1
1.4	Raster scan displays and systems	1
1.5	Video Display devices- LED, OLED, Reflective displays	1

1.6	Touch Panel Technology	1
1.7	Introduction to Multimedia, Authoring and Tools	1
1.8	Popular File Formats	1
1.9	Fundamental Concepts and types of Video	1
1.10	Basics of Digital Audio and its types.	1
Module 2: Line and Shape Drawing Algorithms		8 hours
2.1	DDA Line drawing Algorithm	1
2.2	Bresenham's line drawing algorithm	1
2.3	Midpoint Circle generation algorithm	1
2.4	Bresenham's Circle generation algorithm	1
2.5	Illustration of line drawing and circle drawing algorithms	1
2.6	Ellipse-Generating Algorithms, Properties of Ellipses, Midpoint Ellipse Algorithm	1
2.7	Scan line polygon filling	1
2.8	Boundary filling and flood filling	1
Module 3: Geometric transformations		10 hours
3.1	Basic 2D transformations-Translation and Rotation	1
3.2	Basic 2D transformations- Scaling	1
3.3	Reflection and Shearing	1
3.4	Illustration of 2D Transformations	1
3.5	Composite transformations	1
3.6	General Pivot-Point Rotation	1
3.7	General Fixed-Point Scaling	1
3.8	Matrix representations and homogeneous coordinates	1
3.9	Basic 3D transformations	1
3.10	Illustration of basic 3D transformations	1
Module 4: Transformations and Clipping Algorithms		9 hours
4.1	Window to viewport transformation	1
4.2	Cohen Sutherland Line clipping algorithm	1
4.3	Midpoint subdivision Line clipping algorithm	1
4.4	Liang-Barsky Line Clipping	1
4.5	Nicholl-Lee-Nicholl Line Clipping	1

4.6	Sutherland Hodgeman Polygon clipping algorithm	1
4.7	Weiler Atherton Polygon clipping algorithm	1
4.8	Text Clipping and Exterior Clipping	1
4.9	Practice problems on Clipping algorithms	1
Module 5: Three dimensional graphics and Graphics Functions		8 hours
5.1	Three dimensional viewing pipeline, Projections-Parallel projections	1
5.2	Projections- Perspective projections	1
5.3	Visible surface detection algorithms- Back face detection.	1
5.4	Depth buffer algorithm	1
5.5	Scan line visible surface detection algorithm	1
5.6	A buffer algorithm, Ray-Casting Method	1
5.7	'C' Graphics Functions	1
5.8	Program to initialize the graph and draw a line and circle, program to initialize the graph and draw a rectangle, ellipses and arcs	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Compare the working principle of raster scan systems and random scan systems.
2. How much time is spent scanning across each row of pixels during screen refresh on a raster system with resolution of 1280*1024 and a refresh rate of 60 frames per second?

Course Outcome 2 (CO 2):

1. Rasterize the line using Bresenham's line drawing algorithm with end points (2,3) and (5,8) accepted from the user and implement it using any appropriate programming language.
2. Illustrate how the 4-connected boundary filling approach differs from 8-connected boundary filling and implement it using any appropriate programming language.

Course Outcome 3 (CO 3):

1. Rotate a triangle ABC 45 degree counter clockwise about the pivot point (10,3), where the position vector of the coordinates is given as A(4,1), B(5,2) and C(4,3).
2. Implement the above transformation using any appropriate programming language with user inputs.
3. Illustrate the steps required for a general 3D rotation if the rotation axis is not parallel to any one of the principal axis. The rotation axis is defined by the points P1(x1,y1,z1) and P2(x2, y2, z2). Give its composite matrix representation.

Course Outcome 4 (CO 4):

1. Given a clipping window A(20,20), B(60,20), C(60,40) and D(20,40). Using Cohen Sutherland algorithm, find the visible portion of the line segment joining the points P(40,80) and Q(120,30).
2. Implement Cohen Sutherland clipping algorithm using any appropriate programming language.

Course Outcome 5 (CO 5):

1. Explain scan line algorithm for detecting visible surfaces in an object.
2. Derive the matrix for performing perspective projection and parallel projection.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FOURTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2026

Course Code: B24CSH41

***Course Name: ADVANCED TOPICS IN COMPUTER GRAPHICS
(HONOURS)***

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Out of raster scan system and random scan systems which type of system is beneficial for rendering architectural drawings? Why?
2. What do you understand by the aspect ratio and resolution of a display screen? Compute the width of an image if the image has a height of 2 inches and aspect ratio of 1.5.
3. Write the midpoint circle drawing algorithm?
4. Write the Boundary Filling Algorithm for filling a polygon using four connected approach.
5. A triangle ABC with coordinates A(0,0), B(6,5), C(6,0) is scaled with scaling factors $S_x=2$ and $S_y=3$ about the vertex C(0,0). Find the transformed coordinates.
6. Determine a sequence of basic transformations that is equivalent to x-direction shearing.
7. Find the window to viewport normalization transformation with window lower left corner at (1,1) and upper right corner at (2,6).

8. How does Cohen Sutherland algorithm determine whether a line is visible, invisible or a candidate for clipping based on the region codes assigned to the end points of the line?
9. Differentiate between the object space and image space method for the hidden surface removal of an image.
10. List the various C- Graphics Functions.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the working principle of beam penetration method and shadow mask method with suitable illustrations (8)
(b) Draw the architecture of raster scan display systems and explain its working principle. (6)

OR

12. (a) Explain the working principle of a Refresh CRT monitor with suitable diagrams. (8)
(b) Describe random graphics system with suitable illustrations. (6)
13. (a) Differentiate between boundary fill and flood fill algorithms. (7)
(b) Derive the initial decision parameter of Bresenham's line drawing algorithm and rasterize a line with endpoints (2,2) and (10,10). (7)

OR

14. (a) Write Midpoint circle drawing algorithm and identify the points in the circle with radius as 20 and center at (50,30) using the algorithm. (7)
(b) Illustrate the working principle of scan line polygon filling algorithm. (7)
15. (a) Reflect a triangle ABC about the line $3x-4y+8=0$, where the coordinates of the triangle are given as A(4,1), B(5,2) and C(4,3). (7)
(b) A diamond shaped polygon is located at P(-1,0), Q(0,-2), R(1,0) and S(0,2). Find the transformation matrix which would rotate the triangle by 90 degree counter clockwise about the point Q. Using the transformation matrix, find the coordinates of the rotated polygon. (7)

OR

16. (a) Describe the steps required for a general 3D rotation if the rotation axis is not parallel to any one of the principal axis. The rotation axis is defined by the points $P_1(x_1, y_1, z_1)$ and $P_2(x_2, y_2, z_2)$. Give its composite matrix representation. (8)
(b) Briefly explain Two dimensional transformation with example. (6)

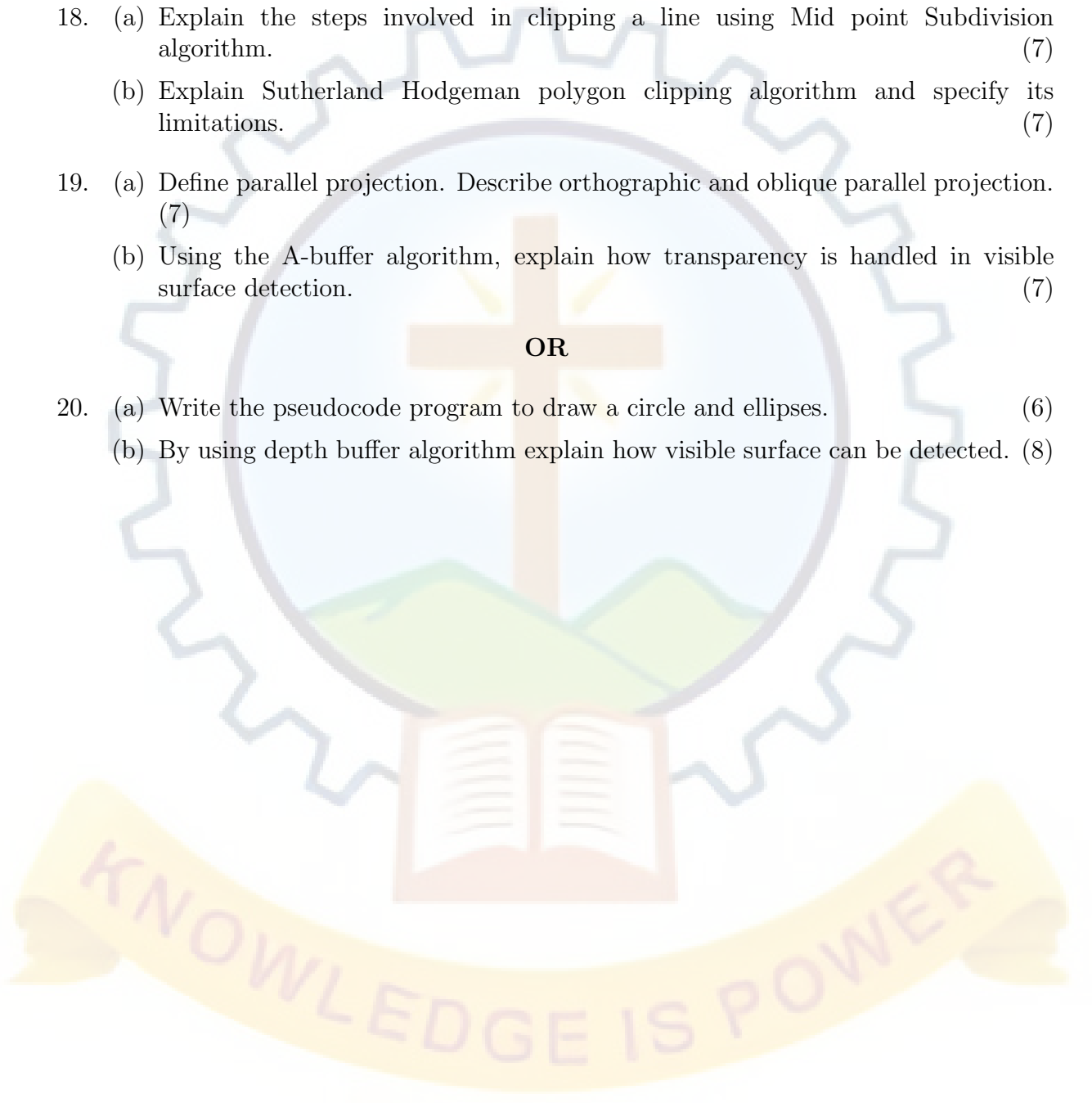
17. (a) Illustrate Weiler – Atherton polygon clipping algorithm. (6)
(b) Given a clipping window A(20,20), B (60,20), C(60, 40) and D(20,40). Using Cohen Sutherland algorithm, find the visible portion of the line segment joining the points P (40,80) and Q(120,30). (8)

OR

18. (a) Explain the steps involved in clipping a line using Mid point Subdivision algorithm. (7)
(b) Explain Sutherland Hodgeman polygon clipping algorithm and specify its limitations. (7)
19. (a) Define parallel projection. Describe orthographic and oblique parallel projection. (7)
(b) Using the A-buffer algorithm, explain how transparency is handled in visible surface detection. (7)

OR

20. (a) Write the pseudocode program to draw a circle and ellipses. (6)
(b) By using depth buffer algorithm explain how visible surface can be detected. (8)



KNOWLEDGE IS POWER

B24CSH42	COMPUTATIONAL FUNDAMENTALS FOR BIOINFORMATICS (Honours)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		

Preamble

This course introduces the fundamental concepts of molecular biology and the essential computational techniques used in bioinformatics. It covers topics such as sequence alignment, biological databases, phylogenetics, and pattern matching, with hands-on experience using R for biological data analysis. The course aims to equip students with the foundational skills required for analyzing and interpreting genomic and proteomic data.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Understand the Basics of Bioinformatics.(Cognitive Knowledge level: Understand)
CO 2	Use various biological databases and apply sequence alignment techniques. Use molecular phylogenetics to identify evolutionary relationships among various biological species.(Cognitive Knowledge level: Apply)
CO 3	Apply the concept of combinatorial pattern matching in bioinformatics.(Cognitive Knowledge level: Apply)
CO 4	Use R language and packages to solve bioinformatics problems,(Cognitive Knowledge level: Apply)
CO 5	Understand protein structure at various levels and utilize computational tools for visualization, prediction, and analysis of protein structures with applications in drug discovery and molecular biology.(Cognitive Knowledge level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	2	1	1	2						2
CO 2	3	3	2	3	3	1					2
CO 3	3	3	2	3	3						2
CO 4	3	3	2	2	3						3
CO 5	3	2	3	3	3	2	1				3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Molecular Biology Primer:

Molecular Biology Primer: Genes, DNAs, RNAs, Proteins, Genomics, Sequencing techniques, Bioinformatics overview and scope. Sequence Alignment: Global and local sequence alignment-dynamic programming algorithms, edit distance, similarity, Needleman Wunsch Algorithm, Smith Waterman Algorithm.

MODULE 2 (9 hours)

Biological Databases and Data Formats:

Biological Databases and Data Formats: Genomic and Sequence Data Formats, GenBank, EMBL-Bank, and DDBJ, PROSITE, NCBI- Database Searching: BLAST, FASTA Phylogenetics: Phylogenetic Tree basics and Construction Methods, UPGMA, Neighbour joining, Parsimonous trees, Additive trees, Bootstrapping.

MODULE 3 (9 hours)

Combinatorial Pattern Matching:

Combinatorial Pattern Matching, Repeat finding, Keyword Trees, Suffix Trees, Heuristic similarity search algorithms, Approximate Pattern Matching.

MODULE 4 (9 hours)

R for Bioinformatics:

Variables, Data types, control flow constructs, String manipulation, Pattern Matching, arrays, lists and hashes, File handling, Programs to handle biological data and parse output files for interpretation, packages for sequence alignment, FASTA, BLAST (Bioconductor, msa, Biostrings etc.)

MODULE 5 (9 hours)

Structural Bioinformatics:

Protein structure basics: Primary, secondary, tertiary, and quaternary structures. Anatomy of DNA; A, B, Z-DNA, DNA bending. Structure prediction, Comparative protein modeling. Protein structure visualization tools (e.g., PyMOL, RasMol, UCSF Chimera). Protein structure databases: PDB, SCOP, CATH Protein structure prediction: Homology modeling, threading, ab initio methods. Structural alignment and comparison. Applications: Drug design, protein-ligand docking (overview), mutation analysis.

Text Books

1. N. C. Jones and P.A. Pevzner, An Introduction to Bioinformatics Algorithms, MIT Press, 2004.
2. Supratim Choudhuri, Bioinformatics for Beginners: Genes, Genomes, Molecular Evolution, Databases and Analytical Tools, Academic Press 1/e, 2014.
3. Robert Gentleman, R Programming for Bioinformatics, CRC Press 1/e, 2009.
4. Philip E. Bourne and Helge Weissig, Structural Bioinformatics, Wiley-Liss 1/e, 2009.
5. Carl Branden & John Tooze, Introduction to Protein Structure, Taylor & Francis, 2/e, 1999.

Reference Books

1. T. K. Attwood and D. J. Parry-Smith, Introduction to Bioinformatics, Pearson Education, 1/e, 2003.
2. B. Junker and F. Schreiber, Analysis of Biological Networks, Wiley Publishers, 1/e, 2007.
3. Y. Sun and J. Han, Mining, Heterogeneous Information Networks - Principles & Methodologies, Morgan & Claypool Publishers, 1/e, 2012.
4. M. E. Dickison et al, Multilayer Social Networks, Cambridge University Press, 1/e, 2016.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Molecular Biology Primer		9 hours
1.1	Genes, DNAs, RNAs, Proteins	1
1.2	Genomics, Sequencing techniques	1
1.3	Bioinformatics overview and scope	1
1.4	Global and local sequence alignment-dynamic programming algorithms	1

1.5	Global and local sequence alignment-dynamic programming algorithms	1
1.6	edit distance	1
1.7	similarity	1
1.8	Needleman Wunsch Algorithm,	1
1.9	Smith Waterman Algorithm	1
Module 2: Biological Databases and Data Formats		9 hours
2.1	Genomic and Sequence Data Formats	1
2.2	GenBank, EMBL-Bank, and DDBJ, PROSITE, NCBI	1
2.3	GenBank, EMBL-Bank, and DDBJ, PROSITE, NCBI	1
2.4	Database Searching: BLAST, FASTA	1
2.5	Phylogenetic Tree basics and Construction Methods	1
2.6	UPGMA	1
2.7	Neighbour joining	1
2.8	Parsimonous trees	1
2.9	Additive trees, Bootstrapping	1
Module 3: Combinatorial Pattern Matching		9 hours
3.1	Combinatorial Pattern Matching	1
3.2	Combinatorial Pattern Matching	1
3.3	Repeat finding	1
3.4	Keyword Trees	1
3.5	Suffix Trees	1
3.6	Heuristic similarity search algorithms	1
3.7	Heuristic similarity search algorithms	1
3.8	Approximate Pattern Matching	1
3.9	Approximate Pattern Matching	1
Module 4: R for Bioinformatics		9 hours
4.1	Variables, Data types	1
4.2	control flow constructs	1

4.3	String manipulation	1
4.4	Pattern Matching	1
4.5	arrays, lists and hashes	1
4.6	File handling	1
4.7	Programs to handle biological data and parse output files for interpretation	1
4.8	packages for sequence alignment	1
4.9	FASTA, BLAST (Bioconductor, msa, Biostrings etc.)	1
Module 5: Structural Bioinformatics		9 hours
5.1	Protein structure basics: Primary, secondary, tertiary, and quaternary structures	1
5.2	Anatomy of DNA; A, B, Z-DNA, DNA bending.	1
5.3	Structure prediction	1
5.4	Comparative protein modeling.	1
5.5	Protein structure visualization tools (e.g., PyMOL, RasMol, UCSF Chimera)	1
5.6	Protein structure databases: PDB, SCOP, CATH	1
5.7	Protein structure prediction: Homology modeling, threading, ab initio methods	1
5.8	Structural alignment and comparison	1
5.9	Applications: Drug design, protein-ligand docking (overview), mutation analysis.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Describe the roles of DNA, RNA, and proteins in genetic information flow.
2. Explain the working of the Needleman-Wunsch algorithm with an example.

Course Outcome 2 (CO 2):

1. Explain the commonly used sequence data formats in bioinformatics. Give examples.
2. Explain the concept of additive trees in phylogenetic analysis.

Course Outcome 3 (CO 3):

1. Describe an algorithm to find repeats in DNA sequences. Why is repeat identification important in phylogenetic analysis?
2. You are given genomic data from five species. Outline a computational strategy using pattern matching techniques to construct a phylogenetic tree.

Course Outcome 4 (CO 4):

1. How can string manipulation techniques be used to implement pattern matching in biological sequences? Illustrate with an example.
2. Write a script to perform exact pattern matching of a given DNA motif within a sequence. How would you modify it for approximate matching?

Course Outcome 5 (CO 5):

1. Compare homology modeling, threading, and ab initio methods for protein structure prediction.
2. How can you retrieve a 3D protein structure from the Protein Data Bank (PDB) and visualize it?

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FOURTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2026

Course Code: B24CSH42

***Course Name: COMPUTATIONAL FUNDAMENTALS FOR
BIOINFORMATICS (HONOURS)***

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Differentiate between DNA and RNA.
2. What does the Needleman-Wunsch algorithm aim to achieve in bioinformatics?
3. What is the role of GenBank in bioinformatics?
4. Define a phylogenetic tree. Why is it important?
5. What is a keyword tree? Mention one use in pattern matching.
6. Define approximate pattern matching. How does it differ from exact matching?
7. Differentiate between an array and a list in R.
8. What is the role of the Biostrings package in R?
9. Differentiate between tertiary and quaternary protein structures.
10. Why is structural alignment important in protein analysis?

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Discuss various DNA/RNA sequencing techniques and their significance in genomics. (7)
(b) Illustrate how edit distance and similarity scoring are used in comparing two sequences. (7)

OR

12. (a) Explain the scope and interdisciplinary nature of bioinformatics. (7)
(b) Write a short note on the significance of sequence alignment in genome annotation. (7)
13. (a) Explain the structure and significance of GenBank, EMBL, and DDBJ databases. (7)
(b) Discuss the working of BLAST and FASTA algorithms for sequence searching. (7)

OR

14. (a) Explain the concept of Parsimony and how it is used to construct phylogenetic trees. (7)
(b) What are additive trees and how are they constructed? Provide an example. (7)
15. (a) Describe the process of repeat finding in biological sequences. Why is it relevant in genome analysis? (7)
(b) Discuss any two heuristic similarity search algorithms. Highlight their advantages and limitations. (7)

OR

16. (a) Explain the concept of combinatorial pattern matching and its applications in genomics. (7)
(b) Compare suffix trees and keyword trees in terms of structure, efficiency, and applications. (7)
17. (a) Describe string manipulation and pattern matching techniques in R with biological examples. (7)
(b) Discuss the importance of file handling in bioinformatics data processing. Include a sample code for reading and parsing a sequence file. (7)

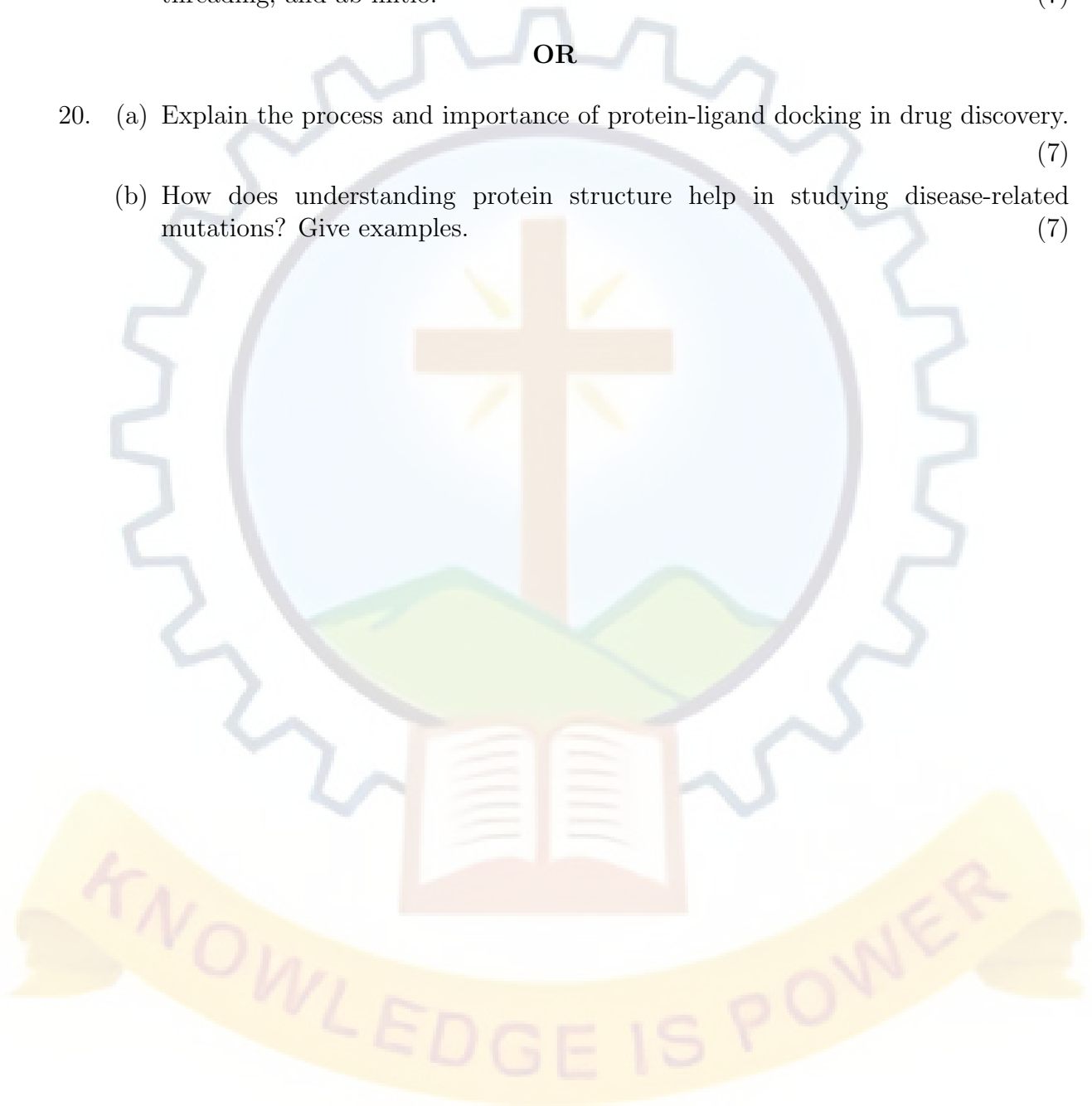
OR

18. (a) Describe the usage of Bioconductor and its packages (msa, Biostrings) in biological sequence analysis. (7)

- (b) Write an R program to read a FASTA file and count the number of sequences. (7)
19. (a) Compare and contrast the functionalities of PyMOL, RasMol, and UCSF Chimera in protein visualization. (7)
- (b) Explain the methods used for protein structure prediction: homology modeling, threading, and ab initio. (7)

OR

20. (a) Explain the process and importance of protein-ligand docking in drug discovery. (7)
- (b) How does understanding protein structure help in studying disease-related mutations? Give examples. (7)



B24CSH43	PRINCIPLES OF PROGRAM ANALYSIS AND VERIFICATION (Honours)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This is the foundational course for awarding B. Tech. Honours in Computer Science and Engineering with specialization in Formal Methods. This course discusses the Foundational Concepts, Methods, Technologies and Tools to ensure reliability and correctness of software systems.

Prerequisites

Discrete Mathematics (B24MA2T03C)

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the concepts and results about Lattices, Chains, Fixed Points, Galois Connections, Monotone and Distributive Frameworks, Hoare Triples, Weakest Preconditions, Loop Invariants and Verification Conditions to perform Analysis and Verification of programs. (Cognitive knowledge level: Understand)
CO 2	Illustrate methods for doing intraprocedural/interprocedural Data flow Analysis for a given Program Analysis problem. (Cognitive knowledge level: Analyse)
CO 3	Formulate an Abstract Interpretation framework for a given Data flow Analysis problem and perform the analysis using the tool WALA. (Cognitive knowledge level: Analyse)
CO 4	Use Kildall's Algorithm to perform Abstract Interpretation of Programs and compare the results obtained by the Algorithm on Monotone and Distributive Frameworks. (Cognitive knowledge level: Apply)
CO 5	Explain the concept of Loop Invariants and use them in Hoare Triple based Weakest Precondition analysis to verify the total correctness of a code segment and use of the tool VCC to specify and verify the correctness of a C Program. (Cognitive knowledge level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	1	2							1
CO 2	2	2	1	3							1
CO 3	2	2	2	2	2						1
CO 4	2	2	2	2							1
CO 5	2	2	2	2	2						1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Mathematical Foundations

Partially Ordered Set, Lattice, Complete Lattice, Construction of Complete Lattices, Chains, Fixed Points, Knaster-Tarski Fixed Point Theorem.

MODULE 2 (7 hours)

Introduction to Program Analysis

The WHILE language, Reaching Definition Analysis, Data Flow Analysis - Equational Approach and Constraint Based Approach, Abstract Interpretation, Algorithm to find the least solutions for the Data Flow Analysis problem.

MODULE 3 (12 hours)

Intraprocedural DataFlow Analysis

Available Expressions Analysis, Reaching Definitions Analysis, Live Variable Analysis, Derived Data Flow Information, Monotone and Distributive Frameworks, Equation Solving - Maximal Fixed Point and Meet Over all Paths solutions.

Interprocedural Data Flow Analysis - Structural Operational Semantics, Intraprocedural versus Interprocedural Analysis, Making Context Explicit, Call Strings as Context, Flow Sensitivity versus Flow Insensitivity, Implementing Interprocedural Data-flow Analysis using the Tool. WALA.

MODULE 4 (8 hours)

Abstract Interpretation

A Mundane Approach to Correctness, Approximations of Fixed Points, Galois Connections, Systematic Design of Galois Connections, Induced Operations, Kildall's Algorithm for Abstract Interpretation.

MODULE 5 (11 hours)

Program Verification

Why should we Specify and Verify Code, A framework for software verification - A core programming Language, Hoare Triples, Partial and Total Correctness, Program Variables and Logical Variables, Proof Calculus for Partial Correctness, Loop Invariants, Verifying code using the tool VCC (Verifier for Concurrent C).

Text Books

1. Flemming Nielson, Henne Nielson and Chris Kankin, Principles of Program Analysis, Springer (1998).
2. Michael Hutch and Mark Ryan, Logic in Computer Science - Modeling and Reasoning about Systems, Cambridge University Press, Second Edition.

Reference Books

1. Julian Dolby and Manu Sridharan, Core WALA Tutorial (PLDI 2010), available online at http://wala.sourceforge.net/files/PLDI_WALA_Tutorial.pdf
2. Ernie & Hillebrand, Mark & Tobies, Stephan (2012), Verifying C Programs: A VCC Tutorial.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Mathematical Foundations		7 hours
1.1	Partially Ordered Set	1
1.2	Lattice	1
1.3	Complete Lattice, Construction of Complete Lattices	1
1.4	Chains 3	1
1.5	Fixed Points	1
1.6	Knaster-Tarski Fixed Point Theorem	1

1.7	Proof of Knaster-Tarski Fixed Point Theorem	1
Module 2: Introduction to Program Analysis		7 hours
2.1	The WHILE language	1
2.2	Data Flow Analysis - Equational Approach	1
2.3	Constraint Based approach	1
2.4	Reaching Definition Analysis	1
2.5	Abstract Interpretation (Lecture 1)	1
2.6	Abstract Interpretation (Lecture 2)	1
2.7	Algorithm to find the least solutions for the Data Flow Analysis problem	1
Module 3: Data flow Analysis		12 hours
3.1	Available Expressions Analysis, Reaching Definitions Analysis	1
3.2	Live Variable Analysis	1
3.3	Derived Data Flow Information	1
3.4	Monotone and Distributive Frameworks	1
3.5	Equation Solving - MFP Solution	1
3.6	Equation Solving - MOP Solution	1
3.7	Structural Operational Semantics	1
3.8	Intraprocedural versus Interprocedural Analysis	1
3.9	Making Context Explicit	1
3.10	Call Strings as Context	1
3.11	Flow Sensitivity versus Flow Insensitivity	1
3.12	Implementing Interprocedural Dataflow Analysis using the Tool WALA	1
Module 4: (Abstract Interpretation		8 hours
4.1	A Mundane Approach to Correctness	1
4.2	Approximations of Fixed Points	1
4.3	Galois Connections	1
4.4	Systematic Design of Galois Connections (Lecture 1)	1

4.5	Systematic Design of Galois Connections (Lecture 2)	1
4.6	Induced Operations	1
4.7	Kildall's Algorithm for Abstract Interpretation (Lecture 1)	1
4.8	Kildall's Algorithm for Abstract Interpretation (Lecture 2)	1
Module 5: Program Verification		11 hours
5.1	Why should we Specify and Verify Code	1
5.2	A framework for software verification - A core programming Language	1
5.3	Hoare Triples (Lecture 1)	1
5.4	THoare Triples (Lecture 2)	1
5.5	Partial and Total Correctness	1
5.6	Program Variables and Logical Variables	1
5.7	Proof Calculus for Partial Correctness	1
5.8	Loop Invariants	1
5.9	Verifying C programs using the tool VCC (Lecture 1)	1
5.10	Verifying C programs using the tool VCC (Lecture 2)	1
5.11	Verifying C programs using the tool VCC (Lecture 3)	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Find a lattice to represent the data states of a given program and propose a sound abstract interpretation framework to do a given analysis on the program.
2. When is an abstract interpretation framework said to be sound? Illustrate with an example.
3. When is an abstract interpretation framework said to be precise? Illustrate with an example.

Course Outcome 2 (CO 2):

1. Illustrate how one can do Intraprocedural Available Expression Analysis on a program.
2. Illustrate how one can do Intraprocedural Reaching Definition Analysis on a program.
3. Illustrate how one can do Intraprocedural Live Variable Analysis on a program.

Course Outcome 3 (CO 3):

1. Illustrate how one can do Interprocedural Data Flow Analysis using the tool WALA

Course Outcome 4 (CO 4):

1. Illustrate the working of Kildall's algorithm to do Intraprocedural Available Expression Analysis on a program
2. Compare the results obtained by applying Kildall's algorithms for Abstract Interpretation in Monotone and Distributive Frameworks.

Course Outcome 5 (CO 5):

1. Illustrate the process of obtaining verification conditions (VCs) using weakest precondition analysis.
2. Explain the concepts of partials and total correctness of programs.
3. Using the tool VCC prove that a given code segment satisfies a given property

MODEL QUESTION PAPER

QP CODE:

Pages: 4

Reg.No.:

Name:

MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM

FOURTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2026

Course Code: B24CSH43

*Course Name: PRINCIPLES OF PROGRAM ANALYSIS AND
VERIFICATION (HONOURS)*

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. What is a complete lattice? Give an example of a complete lattice.
2. Show that every chain is a lattice.
3. Write a program in while language to find the factorial of a number. Explain the statements of your program.
4. Consider a program that calculates x^y through repeated multiplications. Draw the flow graph of the program.
5. What is Available Expression (AE) analysis? Give an application for AE analysis.
6. What is Live variable (LV) analysis? Give an application for LV analysis.
7. Let P be a program analysis problem (like LV, AE, etc.), and (A, F_A, γ_{AC}) and (B, F_B, γ_{BC}) be two abstract interpretations such that A is more abstract than B . Let $\alpha : B \rightarrow A$ and $\gamma : A \rightarrow B$ be the abstraction and concretization functions between A and B . Then, what are the conditions required for α and γ to form a Galois Connection?
8. When is Kildall's algorithm for abstract interpretation guaranteed to terminate? Justify your answer.

9. Is it possible to verify total correctness of a program using Hoare Logic? If yes, how is it possible?
10. Define loop invariant. Show a simple loop with a loop invariant.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) What is an infinite ascending chain in a lattice? Show an example lattice with an infinite ascending chain. Is it possible for a complete lattice to contain an infinite ascending chain? (7 marks)
- (b) State and prove Knaster-Tarski fixed point theorem. (7 marks)

OR

12. (a) Consider the lattice $(\mathbb{N}, \leq)$. Let $f : \mathbb{N} \rightarrow \mathbb{N}$ be a function defined as follows:

$$f(x) = \begin{cases} x + 1, & \text{if } x < 100, \\ x - 1, & \text{if } x > 100, \\ \text{otherwise,} & x = 100. \end{cases}$$

Then, show the following for f :

- i. The set of all fixpoints.
 - ii. The set of all prefixpoints.
 - iii. The set of all postfixpoints. (7 marks)
- (b) Let $(D, \leq)$ be a lattice with a least upper bound for each subset of (D) . Then, prove that every subset of has a greatest lower bound. (7 marks)
13. (a) With a suitable example, explain the equational approach in Data Flow Analysis. (7 marks)
 - (b) With a suitable example, explain how you obtain the collecting semantics of a program point. (7 marks)

OR

14. (a) With an example, explain the Constrained Based Approach in Data Flow Analysis. (7 marks)
- (b) Discuss the properties of an algorithm to solve the problem of computing the least solution to the program analysis problems in Data Flow Analysis. (7 marks)

15. (a) Using Intraprocedural Reaching Definition Analysis, find the assignments killed and generated by each of the blocks in the program

```
[x := 5]1;  
[y := 1]2;  
while [x > 1]3 do  
    ([y := x * y]4; [x := x - 1]5)
```

(7 marks)

- (b) Analyse the following program using Intraprocedural Very Busy Expression analysis

```
if [a > b]1 then  
    ([x := b - a]2; [y := a - b]3)  
else  
    ([y := b - a]4; [x := a - b]5)
```

(7 marks)

OR

16. (a) Find Maximal Fixed Point (MFP) solution for the program

```
[x := a + b]1;  
[y := a * b]2;  
while [y > a + b]3 do  
    ([a := a + l]4; [x := a + b]5)
```

(7 marks)

- (b) With examples, explain the difference between flow sensitive and flow insensitive analysis. (7 marks)

17. (a) Prove that (L, α, γ, M) is an adjunction if and only if (L, α, γ, M) is a Galois connection. (7 marks)

- (b) Prove that if $\alpha : L \rightarrow M$ is completely additive, then there exists $\gamma : M \rightarrow L$ such that (L, α, γ, M) is a Galois connection. Similarly, if $\gamma : M \rightarrow L$ is completely multiplicative, then there exists $\alpha : L \rightarrow M$ such that (L, α, γ, M) is a Galois connection. (7 marks)

OR

18. (a) Show that if $(L_i, \alpha_i, \gamma_i, M_i)$ are Galois connections and $\beta_i : V_i \rightarrow L_i$ are representation functions, then

$$((\alpha_1 \circ \beta_1) \Rightarrow (\alpha_2 \circ \beta_2))(\Rightarrow) = \alpha_2 \circ ((\beta_1 \Rightarrow \beta_2)(\Rightarrow)) \circ \gamma_1$$

(7 marks)

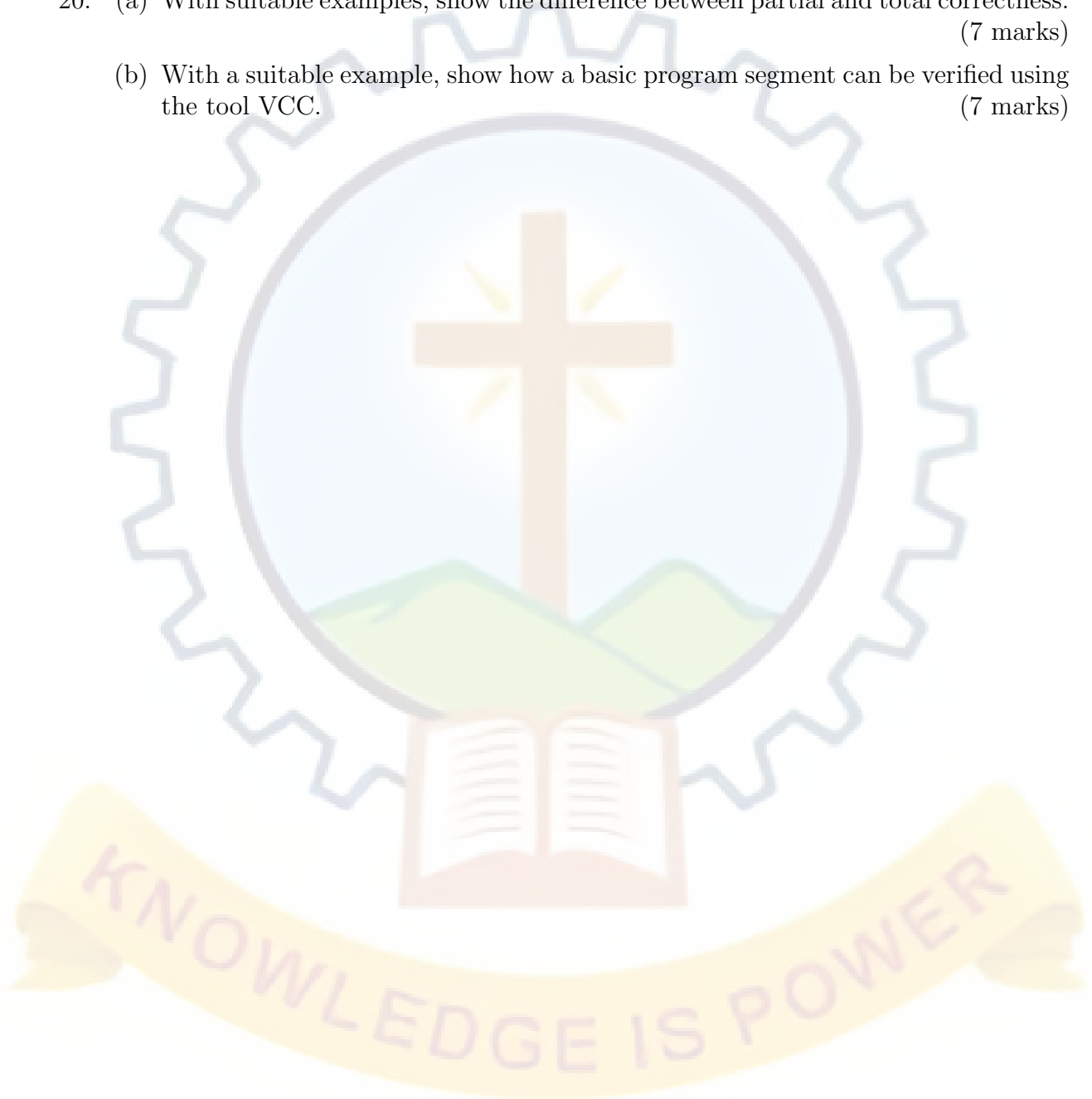
- (b) Briefly explain Kildall's algorithm for abstract interpretation. (7 marks)

19. (a) Briefly explain the need of specification and verification of code. (7 marks)

- (b) Argue that Hoare Logic is sound. When Hoare Logic is complete? Let APB be a Hoare triple such that Hoare Logic is complete for the program P. Then, is it always possible to check the validity of the Hoare Triple? If not, what is the difficulty? (7 marks)

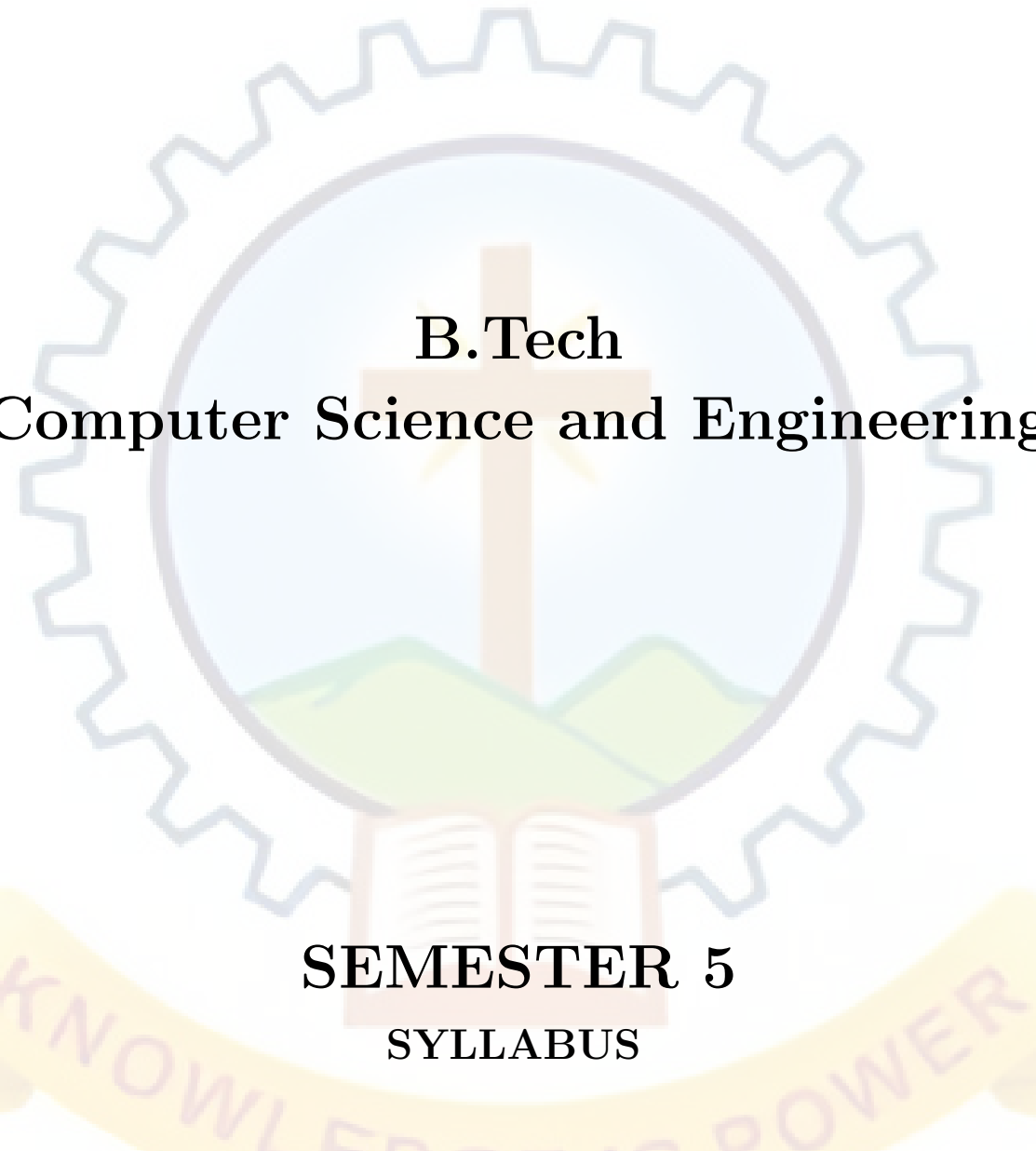
OR

20. (a) With suitable examples, show the difference between partial and total correctness. (7 marks)
- (b) With a suitable example, show how a basic program segment can be verified using the tool VCC. (7 marks)



MAR ATHANASIOUS COLLEGE OF ENGINEERING

Government Aided, Autonomous Institution
Kothamangalam, Kerala, India

The logo of Mar Athanasius College of Engineering is a circular emblem. It features a large gear-like outer border. Inside the circle, there is a light blue sky with a white cross in the center. Below the cross are green hills. At the bottom of the circle is an open book. A yellow banner with the text 'KNOWLEDGE IS POWER' in purple capital letters curves around the bottom of the emblem.

B.Tech Computer Science and Engineering

SEMESTER 5 SYLLABUS

SEMESTER 5

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24CS3T01	MICROPROCESSOR SYSTEMS	3-1-0-5	4	4
B	B24CS3T02	WEB PROGRAMMING	3-1-0-5	4	4
C	B24CS3T03	COMPILER DESIGN	3-1-0-5	4	4
D	B24CS3T04	ALGORITHM ANALYSIS AND DESIGN	3-1-0-5	4	4
E	B24CS3T05	ARTIFICIAL INTELLIGENCE	2-1-0-4	3	3
F	B24CS3P1x	PROGRAMME ELECTIVE I	2-1-0-4	3	3
G	B24CS3L07	SYSTEM SOFTWARE AND COMPILER DESIGN LAB	0-0-3-2	3	2
H	B24CS3L08	WEB PROGRAMMING LAB	0-0-3-2	3	2
M	B24CSM5x	MINOR	3-1-0-5	4	4
N	B24CSH5x	HONOURS	3-1-0-5	4	4
TOTAL				36	26

PROGRAMME ELECTIVE I

COURSE NO.	COURSES
B24CS3P11	PROGRAMMING PARADIGMS (Bucket-I)
B24CS3P12	NUMBER THEORY (Bucket-II)
B24CS3P13	INTERNET OF THINGS (General-I)
B24CS3P14	DISTRIBUTED COMPUTING (General-II)

MINOR

COURSE NO.	COURSES
B24CSM51	DATABASE PROGRAMMING
B24CSM52	CONCEPTS IN DEEP LEARNING
B24CSM53	CLIENT SERVER SYSTEMS

HONOURS

COURSE NO.	COURSES
B24CSH51	ADVANCED CONCEPTS IN COMPUTER VISION
B24CSH52	MACHINE LEARNING IN COMPUTATIONAL BIOLOGY
B24CSH53	PRINCIPLES OF MODEL CHECKING

B24CS3T01	MICROPROCESSOR SYSTEMS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course builds core competency in microprocessor architecture and programming, covering 8085/8086 fundamentals, assembly language, memory and I/O interfacing, interrupts/DMA, and contemporary MCU practice (STM32), with extensions to IoT, wireless and RTOS concepts.

Prerequisites

Logic System Design, Computer Organization and Architecture, Basic Electronics, C Programming.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Understand the architecture, memory organization, instruction formats, and addressing modes of 8085 and 8086 microprocessors.(Cognitive Knowledge Level: Understand).
CO 2	Apply the 8086 instruction set and assembler directives to develop assembly language programs for basic computational and control tasks. (Cognitive Knowledge Level: Apply).
CO 3	Apply stack operations, interrupt handling techniques, and interfacing methods to design simple microprocessor-based applications. (Cognitive Knowledge Level: Apply).
CO 4	Use STM32 microcontroller peripherals to build embedded applications involving digital I/O, sensors, ADC/DAC operations, displays, and timers. (Cognitive Knowledge Level: Apply).
CO 5	Apply IoT protocols, wireless communication interfaces, and basic RTOS features to develop small-scale IoT and real-time embedded systems. (Cognitive Knowledge Level: Apply).

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1							1		3
CO 2	2	3	2	2	2	1			1		3
CO 3	2	3	2	3	3	2		3	3	1	3
CO 4	2	3	2	3	3	3		3	3	2	3
CO 5	2	3	2	3	3	3	2	3	3	2	3

Assessment Pattern

Bloom's Category	Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

8085/8086 Fundamentals:

8085 Microprocessor (basic architecture overview, no need to study the diagram).8086 Microprocessor – Architecture and signals, Physical Memory organization, Minimum and maximum mode of 8086 system and timings. Comparison of 8086 and 8088. Machine language instruction format. Addressing Modes of 8086.

MODULE 2 (8 hours)

8086 Instruction Set & Assembly:

Instruction set – data copy/transfer, arithmetic, logical, string manipulation, branch instructions; unconditional and conditional branch; flag manipulation and processor control instructions. Assembler directives and operators. Assembly Language Programming with 8086..

MODULE 3 (8 hours)

Stack, Interrupts & Interfacing Basics:

Stack and Interrupts: Stack structure of 8086; programming using stack. Interrupt structure of 8086 microprocessor, processing of vectored and non vectored interrupts, Handling multiple interrupts, Interfacing: interrupt controller chip, Interfacing of memory chips, address allocation technique and decoding; .

MODULE 4 (14 hours)

STM32 Microcontroller Overview and Peripheral Programming:

Introduction to STM32 Family, STM32U575 Features and Specifications, Introduction to Integrated Development Environment and HAL, Writing, and Debugging Your First Program(LED Interfacing), Interfacing Seven-Segment Display, LCD Display, and Matrix Keypad, Relay Interfacing, Analog to Digital Conversion: Potentiometer, temperature sensor, LDR, Microphone, Digital to Analog Conversion: Simple DAC Output Generation,

Generating a Sine Wave, Audio Signal Generation. Timer and Counter Applications: Basic Timer Configuration, Timers as Counters, Timer-Based Real-Time Clock (RTC) .

MODULE 5 (8 hours)

IoT, Wireless Communication, and RTOS:

IoT Architecture, Protocols (MQTT), IoT Security Principles and Common Threats
Wireless Communication: Interfacing GSM (Call, SMS, Internet), Bluetooth
Communication Basics, LoRa Communication Basics and Applications, Designing an
IoT-Based Home Automation System, Introduction to RTOS Concepts, FreeRTOS with
STM32: Task Creation, Scheduling, and Management, RTOS Timers, Delays, and RTC

Text Books

1. R. S. Gaonkar, Microprocessor Architecture, Programming and Applications with the 8085, 6/e, Pearson.
2. B. Brey, The Intel Microprocessors: 8086/8088, 80186, 80286, 80386, 80486, Pentium and Beyond, 8/e, Pearson.
3. Douglas V. Hall (with S. S. Sriram), Microprocessors and Interfacing: Programming and Hardware, 2/e, McGraw-Hill.
4. Joseph Yiu, The Definitive Guide to ARM® Cortex-M23 and Cortex-M33 Processors, 2/e, Newnes (for STM32U5/Cortex-M33).
5. STMicroelectronics, STM32U5 Series Reference Manual (RM0456 or latest) + STM32Cube HAL Driver User Manual (official docs).

Reference Books

1. A. P. Godse & D. A. Godse, Microprocessor and Interfacing, Technical Publications.
2. M. Rafiquzzaman, Microcomputer Theory and Applications, Wiley.
3. Kamal Hyder & Bob Perrin, Embedded Systems Design with the Atmel AVR Microcontroller (for interfacing concepts).
4. Brian Amos, Hands-On RTOS with Microcontrollers: Building Real-Time Systems Using FreeRTOS, STM32 and NXP, Packt.
5. Kevin Townsend et al., Getting Started with Bluetooth Low Energy, O'Reilly.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: 8085/8086 Fundamentals		7 hours
1.1	8085 and 8086 overview – basic architecture, signals	1
1.2	Physical memory organization and bus structure	1
1.3	Minimum and maximum modes of 8086; timings	1
1.4	8086 vs 8088 features; machine language format; addressing modes	1
1.5	Instruction cycle and execution overview	1
1.6	Common applications and limitations	1
1.7	Revision and case discussion	1
Module 2: 8086 Instruction Set and Assembly		8 hours
2.1	Data transfer and arithmetic instructions	1
2.2	Logical, string and flag manipulation instructions	1
2.3	Branch and loop instructions; conditional flag logic	1
2.4	Assembler directives and operators	1
2.5	Procedures, parameter passing, and stacks	1
2.6	Program examples: array, sum, and factorial	1
2.7	TOPIC 7Debugging and testing assembly programs	1
2.8	Practical session discussion	1
Module 3: Stack, Interrupts Interfacing Basics		8 hours
3.1	Stack organization and stack-based programming	1
3.2	8086 interrupt structure and IVT and Vectored vs non-vectored interrupts	1
3.3	8259A PIC operation and initialization	1
3.4	Memory chip interfacing – address decoding	1
3.5	Memory chip interfacing – address decoding	1
3.6	I/O interfacing concepts	1
3.7	Address allocation techniques	1

3.8	Timing analysis and case study	1
Module 4: STM32 Microcontroller Overview and Peripheral Programming		14 hours
4.1	Introduction to STM32 family, series taxonomy, ecosystem	1
4.2	STM32U575 features and specifications	1
4.3	IDE setup and HAL introduction	1
4.4	First program: LED GPIO write debugging workflow	1
4.5	Seven-segment display interfacing	1
4.6	LCD (16×2) interfacing and basic APIs	1
4.7	Matrix keypad scanning and debouncing	1
4.8	Relay interfacing and protection (transistor/diode)	1
4.9	ADC basics potentiometer reading	1
4.10	ADC with sensors: temperature, LDR, microphone	1
4.11	DAC basics simple analog output generation	1
4.12	DAC applications: sine-wave and audio signal generation	1
4.13	Timer configuration using timers as counters	1
4.14	Timer-based RTC implementation and validation	1
Module 5: IoT, Wireless Communication, and RTOS		8 hours
5.1	IoT architecture, protocols (MQTT, CoAP) — concepts, layers, publish-subscribe, request-response	1
5.2	IoT security — principles, common threats, basic mitigations	1
5.3	GSM interfacing — call, SMS, internet data (AT commands overview)	1
5.4	Bluetooth communication — pairing, services, simple data exchange	1
5.5	LoRa communication — basics, range, applications, simple link setup	1
5.6	IoT home-automation design — end-to-end block diagram, workflow	1
5.7	RTOS essentials — concepts, FreeRTOS overview, tasks scheduling	1
5.8	FreeRTOS on STM32 — task management, timers, delays, RTC usage	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. How would the performance and memory access pattern of an 8086 system change if it were configured in maximum mode instead of minimum mode?
2. Given an embedded system that must interface both 8-bit and 16-bit peripherals, justify whether choosing 8088 or 8086 would be more efficient and why.

Course Outcome 2 (CO 2):

1. Design a short assembly routine to check whether a character in memory is a vowel and explain which addressing mode improves efficiency.
2. Suppose an industrial controller needs periodic sensor sampling—show how string manipulation and loop instructions could reduce instruction count compared to conventional branching.

Course Outcome 3 (CO 3):

1. A robotic arm must handle emergency-stop inputs from multiple sensors. Explain how you would use 8259A PIC and interrupt vectors of 8086 to prioritize these signals.
2. If a system uses both memory-mapped and isolated I/O, discuss how address decoding should be modified to avoid bus contention and false triggering.

Course Outcome 4 (CO 4):

1. In a home-automation scenario, describe how you would combine ADC and DAC features of STM32 to sense ambient light and control lamp brightness automatically.
2. Given a requirement to display data on both LCD and seven-segment displays simultaneously, propose a GPIO allocation and timing strategy to avoid display flicker.

Course Outcome 5 (CO 5):

1. Imagine you are developing a remote weather-monitoring system. Explain how you would use MQTT over GSM with STM32 to transmit periodic sensor readings reliably.
2. A factory-monitoring RTOS task must log data every second without drift. How can FreeRTOS timers and RTC synchronization be configured to guarantee precise periodic execution?

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CS3T01

Course Name: MICROPROCESSOR SYSTEMS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Compare the main architectural differences between 8085 and 8086.
2. List and explain any two addressing modes of 8086 with examples.
3. What is the role of assembler directives in assembly language programming?
4. Distinguish between conditional and unconditional branch instructions in 8086.
5. Define stack and explain its role in subroutine execution.
6. What is the difference between vectored and non-vectored interrupts?
7. List any four features of STM32U575 microcontroller.
8. What is the purpose of the HAL library in STM32 programming?
9. Differentiate between MQTT and CoAP protocols used in IoT systems.
10. What are the basic principles of IoT security?

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the architecture of 8086 microprocessor with a neat diagram and describe the function of each block. (7)
- (b) Describe the minimum and maximum mode operations of 8086 with timing diagrams. (7)

OR

12. (a) A legacy product must be migrated from 8088 to 8086 without changing peripheral boards. Propose a migration plan addressing memory map, timing, and bus behavior. (7)
- (b) A real-time meter needs deterministic memory access. Compare segmented addressing vs a “flat” illusion created by linker/loader choices, and recommend one with reasons. (7)
13. (a) Describe the different categories of instructions in the 8086 instruction set with examples. (7)
- (b) Write and explain an assembly program to find the smallest number in an array. (7)

OR

14. (a) A firmware routine that sorts sensor readings is too slow. Re-implement the hot path using addressing modes and string operations to reduce instruction count; justify each change. (7)
- (b) A data-logger must convert mixed-case ASCII to uppercase in-place and compute a checksum in one pass. Design and explain the routine and flag usage. (7)
15. (a) Explain the structure of the 8086 stack and demonstrate stack-based arithmetic operations. (7)
- (b) Discuss the concept of interrupt vector table and illustrate how multiple interrupts are prioritized. (7)

OR

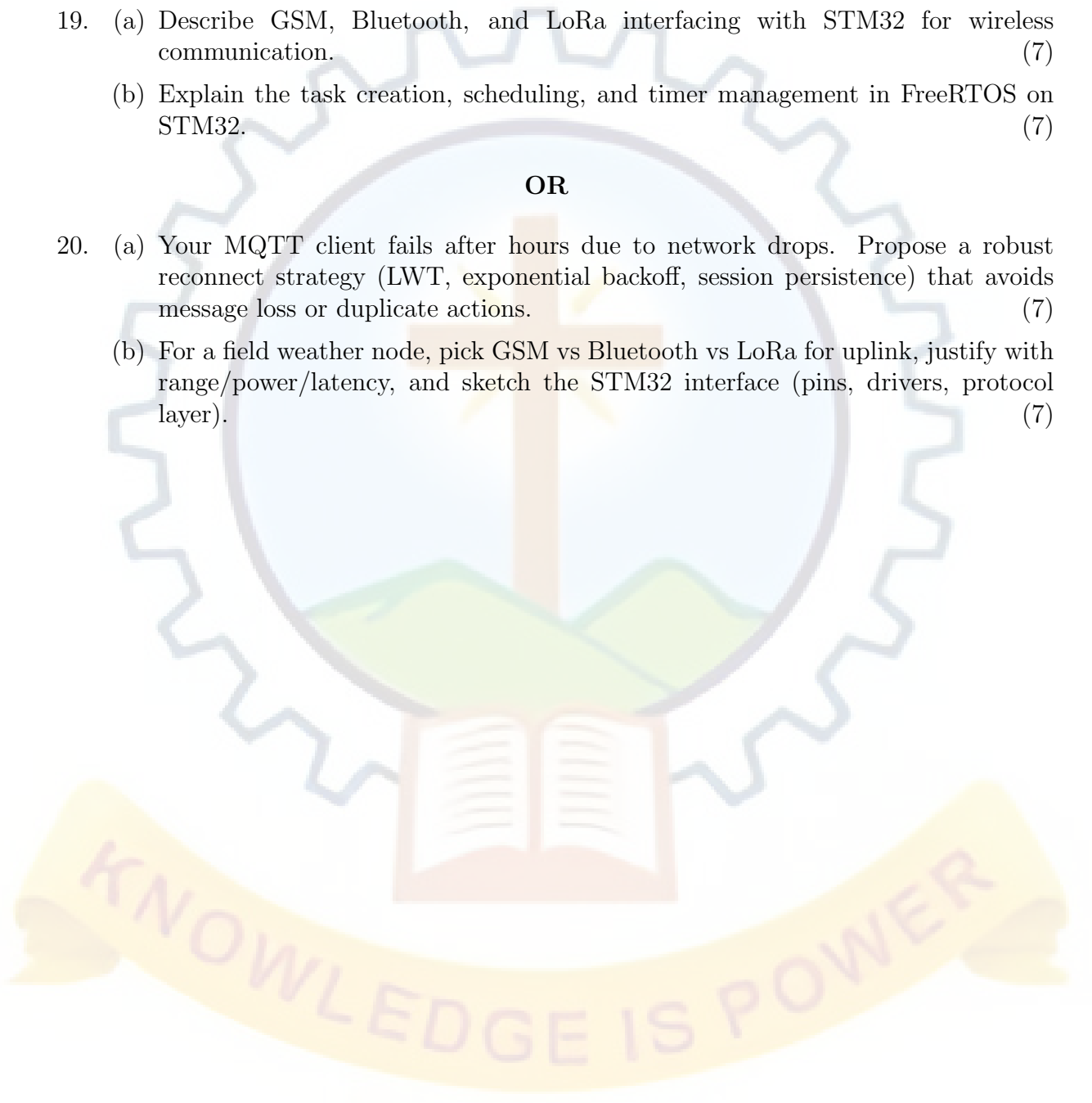
16. (a) Design an interrupt scheme for three sensors (critical, warning, info) so that the critical event always preempts others. Include vector assignments and PIC settings. (7)
- (b) You need to add a 16-KB RAM and 8-KB ROM to an already crowded map. Draw a high-level address allocation & decoding plan that avoids bus contention and meets timing. (7)
17. (a) Explain the procedure to configure STM32 IDE and write your first LED blinking program. (7)
- (b) Describe the interfacing of seven-segment display and LCD display with STM32. (7)

OR

18. (a) You must display sensor values on seven-segment and LCD concurrently with minimal pins. Propose a multiplexing/driver plan and timing so neither flickers. (7)
- (b) Draft a bring-up checklist for a new STM32U575 board: clock tree, pin muxing, HAL init, and a minimal LED + UART sanity test—explain the order and failure signals. (7)
19. (a) Describe GSM, Bluetooth, and LoRa interfacing with STM32 for wireless communication. (7)
- (b) Explain the task creation, scheduling, and timer management in FreeRTOS on STM32. (7)

OR

20. (a) Your MQTT client fails after hours due to network drops. Propose a robust reconnect strategy (LWT, exponential backoff, session persistence) that avoids message loss or duplicate actions. (7)
- (b) For a field weather node, pick GSM vs Bluetooth vs LoRa for uplink, justify with range/power/latency, and sketch the STM32 interface (pins, drivers, protocol layer). (7)



KNOWLEDGE IS POWER

B24CS3T02	WEB PROGRAMMING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		2024

Preamble

This course provides a comprehensive understanding of web programming concepts and technologies used in the development of modern web applications. It introduces HTML5 and XML for structuring web content and data, CSS for styling, layout, and responsive web design, and JavaScript for implementing client-side scripting, dynamic behavior, and event handling. The course also covers server-side programming using PHP, database connectivity with MySQL, and fundamental web architecture concepts including HTTP transactions and multi-tier application design. In addition, the course introduces Node.js and Express for developing scalable, asynchronous server-side web applications. The course enables students to design, develop, and deploy secure, responsive, and data-driven web applications.

Prerequisites

Basic programming skills in any structured programming language (such as C /Python/ Java), Basics of Database Management Systems

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Apply HTML5 and CSS concepts to design structured, semantically meaningful, responsive, and visually appealing web pages. (Cognitive Knowledge Level: Apply)
CO 2	Apply JavaScript to add functionality to web pages. (Cognitive Knowledge Level: Apply)
CO 3	Develop web applications using PHP and MySQL. (Cognitive Knowledge Level: Apply)
CO 4	Apply Node.js and Express to build web servers and perform CRUD operations. (Cognitive Knowledge Level: Apply)
CO 5	Apply React and REST API concepts to build component-based user interfaces and integrate CRUD operations using HTTP methods. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	2	1	1	2		2	3		1
CO 2	2	2	2	3	2	2	2	3	3	1	3
CO 3	2	3	3	3	3	3	3	3	3	3	3
CO 4	2	3	3	3	3	3	3	3	3	3	3
CO 5	2	3	3	3	3	3	3	3	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (% Marks)	Test 2 (% Marks)	
Remember			
Understand	40	40	40
Apply	60	60	60
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Introduction to Web Designing

Introduction To Web Designing and HTML5, Basic Structure for HTML, Basic HTML tags – Headings, Linking, Images, Special Characters and Horizontal Rules, Lists, Tables, Forms, Internal Linking, meta elements, New HTML5 Form input Types, input and data list elements, autocomplete Attribute, Page-Structure Elements. Style With Css Introduction to CSS- Introduction, Inline Styles, Embedded Style Sheets, Linking External Style Sheets. Responsive Web page with HTML and CSS

MODULE 2 (9 hours)

Javascript

JavaScript in browser, Basic JavaScript instructions: statements, comments, variables, data types, arrays, expressions, operators; functions, methods and objects: function, anonymous function, variable scope. Decision making and Loops: if statement, if...else statement, switch statement; loops: key loop concepts, for loops, while loops, do while loops. Event handling: using event listeners, using parameters with event listeners, the event object, event delegation, user interface events.

MODULE 3 (9 hours)

Php And MySQL Database

Web Servers: Introduction, HTTP Transactions, Multitier Application Architecture, Client-Side Scripting versus Server-Side Scripting. PHP: Starting to script on server side, Arrays, functions and forms, advanced PHP. Working with Databases: Basic commands with PHP examples, Connection to server, Database APIs, Accessing MySQL in PHP, PHP MyAdmin and database bugs.

MODULE 4 (9 hours)

Node.Js

Node.Js Server Side Scripting with Node.js: Introduction to Node.js , features of Node, asynchronous callbacks. Hello World in Node, package.json, modules, Built-in Modules: FS Module, HTTP Module, Events; Node Package Manager (npm), web server using http, Node.js with Express, middleware, routing in Express, CRUD operations in Express, web server using Express.

MODULE 5 (9 hours)

React and REST API

Rendering HTML : React - ReactJS Foundations : The Philosophy of React, What is a component? Built- in components, Userdefined components - Types of components, Function Components, Differences between Function and Class Components Introduction to REST services; principles of REST architecture and resource-based design. HTTP Methods and API Design: HTTP methods including GET (retrieve data), POST (create resource), PUT (update resource), and DELETE (remove resource); mapping of CRUD operations to HTTP methods; designing REST APIs for real-world applications with proper endpoints and structure.

Text Books

1. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, “Internet and World Wide Web How To Program”, 5/E, Pearson Education, 2018.
2. Jon Duckett, “HTML and CSS: Design and Build Websites”, Wiley, 1st Edition 2011.
3. Jon Duckett, “JavaScript and JQuery: Interactive Front-End Web Development”, Wiley, 1st Edition 2013.
4. P.J. Deitel & H.M. Deitel, “Internet and World Wide Web How to Program”, 4th Edition, Pearson Education.
5. HTML 5 Black Book, Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP and jQuery, 2/e, 2016.
6. Greg Lim, “Beginning Node.js, Express and MongoDB Development”, 1st Edition 2019.

Reference Books

1. Alex Young, Marc Harter, “Node.js in Practice”, Manning.
2. Jon Krol, “Web Development with MongoDB and Node.js”, Packt.
3. Krishna Rungta, “Node JS: Learn in One Day”.
4. Nixon, Robin. “Learning PHP, MySQL, JavaScript, and CSS”, O’Reilly Media, 2012.
5. Web Applications: Concepts and Real World Design, Knuckles, Wiley-India.
6. Pritma Jashnani, “A Hand Book On Web Development: From Basics of HTML to JavaScript and PHP”, Notion Press, 1/e, 2022.
7. Jon Duckett, “PHP & MySQL: Server Side Web Development”, John Wiley & Sons Inc., 1/e, 2022.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: INTRODUCTION TO WEB DESIGNING		9 hours
1.1	Introduction to Web Designing and HTML5	1
1.2	Basic structure of HTML	1
1.3	Basic HTML tags: headings, linking, images, special characters, horizontal rules	1
1.4	Lists, tables, forms, internal linking, meta elements	1
1.5	New HTML5 form input types, input and datalist elements, autocomplete attribute	1
1.6	Page-structure elements in HTML5	1
1.7	Introduction to CSS: inline styles, embedded style sheets, external style sheets	1
1.8	Styling and layout using CSS	1
1.9	Responsive web pages using HTML and CSS	1
Module 2: JAVASCRIPT		9 hours
2.1	JavaScript in browser, introduction to JavaScript	1
2.2	Basic JavaScript instructions: statements, comments, variables	1
2.3	Data types, arrays, expressions and operators	1
2.4	Functions, anonymous functions, variable scope	1
2.5	Objects and methods in JavaScript	1
2.6	Decision making: if, if-else, switch statements	1
2.7	Loops: key loop concepts, for, while, do-while loops	1
2.8	Event handling: event listeners, parameters, event object	1
2.9	Event delegation and user interface events	1
Module 3: PHP AND MYSQL DATABASE		9 hours
3.1	Web servers: introduction, HTTP transactions	1
3.2	Multitier application architecture	1

3.3	Client-side scripting vs server-side scripting	1
3.4	Introduction to PHP, starting server-side scripting	1
3.5	PHP arrays, functions and forms	1
3.6	Advanced PHP concepts	1
3.7	Working with databases: basic commands with PHP examples	1
3.8	Database connectivity: connection to server, database APIs, accessing MySQL in PHP	1
3.9	PHPMyAdmin and database bugs	1
Module 4: NODE.JS		9 hours
4.1	Introduction to Node.js, features of Node.js	1
4.2	Asynchronous callbacks in Node.js	1
4.3	Hello World in Node.js, package.json	1
4.4	Modules in Node.js	1
4.5	Built-in modules: FS module, HTTP module, Events	1
4.6	Node Package Manager (npm)	1
4.7	Web server using HTTP module	1
4.8	Node.js with Express: middleware and routing	1
4.9	CRUD operations and web server using Express	1
Module 5: REACT AND REST API		9 hours
5.1	Introduction to React, rendering HTML using React	1
5.2	ReactJS foundations, philosophy of React	1
5.3	Components in React: built-in and user-defined components	1
5.4	Types of components: function components, class components and their differences	1
5.5	Introduction to REST services and resource-based design	1
5.6	Principles of REST architecture	1
5.7	HTTP methods: GET, POST, PUT, DELETE	1
5.8	Mapping CRUD operations to HTTP methods	1
5.9	Designing REST APIs with proper endpoints and structure	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the difference between internal linking and external linking and give examples
2. Design an HTML5 form to collect the following details from a user: Name, Email, Phone Number, Date of Birth, Gender (radio buttons), and a Short Bio(textarea). Include proper labels and placeholders.

Course Outcome 2 (CO 2):

1. Web application requires validating user input and performing calculations dynamically. Explain how JavaScript variables, data types, and operators can be used to achieve this functionality.
2. Write a JavaScript program that does the following: Create an array of 5 students' names. Prompt the user to enter a name. Check if the entered name exists in the array. If it exists, display "Welcome [Name]!". If not, display "Name not found!". Sort the array alphabetically and print it. Use at least one function and one loop

Course Outcome 3 (CO 3):

1. Write a JavaScript program that does the following: Create an array of 5 students' names. Prompt the user to enter a name. Check if the entered name exists in the array. If it exists, display "Welcome [Name]!". If not, display "Name not found!". Sort the array alphabetically and print it. Use at least one function and one loop
2. Explain database handling in PHP. Discuss database APIs, accessing MySQL, and the role of PHPMyAdmin in database management.

Course Outcome 4 (CO 4):

1. Explain modules in Node.js and the difference between built-in and custom modules.
2. What is Node Package Manager (npm)? Explain its functions and how it is used in Node.js applications.

Course Outcome 5 (CO 5):

1. Design a REST API for a real-world application (e.g., student management system). Define endpoints and explain their functionality
2. Design a simple React component to render HTML content. Explain the structure and working of the component.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CS3T02

Course Name: WEB PROGRAMMING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. How does the use of meta elements improve the performance and accessibility of a web page? Give suitable examples.
2. Differentiate between inline, embedded, and external CSS with examples.
3. Explain different data types in JavaScript with examples.
4. Describe event handling in JavaScript and the role of event listeners.
5. Differentiate between client-side scripting and server-side scripting
6. Explain how PHP connects to a MySQL database with basic steps.
7. What are the key features of Node.js? Explain asynchronous callbacks
8. Explain the role of Express.js in Node.js applications.
9. What is a component in React? Differentiate between function and class components.
10. Explain REST architecture and the use of HTTP methods (GET, POST, PUT, DELETE).

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Design a webpage that displays the below table:

Food Item	Recommended Intake			
	age<15		age>15	
	gm	Kcal	gm	Kcal
Cerials	1000	2000	750	1760
NonCerials	450	800	350	600

(7)

- (b) Explain the difference between internal linking and external linking and give examples. (7)

OR

12. (a) Design an HTML5 form to collect the following details from a user: Name, Email, Phone Number, Date of Birth, Gender (radio buttons), and a Short Bio (textarea). Include proper labels and placeholders. (7)
- (b) Enhance the form using HTML5 input types (email, date, number) and autocomplete attribute where applicable. Design a responsive web page using HTML and CSS. Explain the concepts and techniques used to achieve responsiveness. (7)
13. (a) Explain JavaScript objects and methods. How are they used in real-world web applications. (7)
- (b) Describe event handling in JavaScript. Explain event listeners, event objects, and event delegation with examples (7)

OR

14. (a) A web application requires validating user input and performing calculations dynamically. Explain how JavaScript variables, data types, and operators can be used to achieve this functionality. (7)
- (b) Write a JavaScript program that does the following: Create an array of 5 students' names. Prompt the user to enter a name. Check if the entered name exists in the array. If it exists, display "Welcome [Name]!". If not, display "Name not found!". Sort the array alphabetically and print it. Use at least one function and one loop (7)
15. (a) Explain database handling in PHP. Discuss database APIs, accessing MySQL, and the role of PHPMyAdmin in database management. (7)
- (b) Explain PHP scripting in detail. Discuss arrays, functions, and form handling in PHP with examples.. (7)

OR

16. (a) A college wants to develop a simple student registration system using PHP and MySQL. Design a web application that provides an HTML form to collect student details such as Name, Roll Number, and Course. When the user submits the form, the PHP program should validate the input fields, establish a connection to a MySQL database, and store the data in a table named students. After successful insertion, the application should retrieve all records from the database and display them in a structured tabular format on the webpage. Include the necessary SQL query to create the table and write the PHP code for database connection, data insertion, and retrieval.. (7)
- (b) Write a PHP script to demonstrate the use of arrays and functions by storing a list of numbers and finding their sum and average. (7)
17. (a) Explain the role of modules in Node.js. Discuss built-in modules such as FS, HTTP, and Events with examples. (7)
- (b) Explain how to create a web server using Node.js and Express. Discuss middleware, routing, and CRUD operations.. (7)

OR

18. (a) Explain modules in Node.js and the difference between built-in and custom modules. (7)
- (b) What is Node Package Manager (npm)? Explain its functions and how it is used in Node.js applications. (7)
19. (a) Describe REST architecture. Explain its principles and resource-based design approach.. (7)
- (b) Design a simple React component to render HTML content. Explain the structure and working of the component. (7)

OR

20. (a) Design a REST API for a real-world application (e.g., student management system). Define endpoints and explain their functionality (7)
- (b) Design a simple React component to render HTML content. Explain the structure and working of the component. (7)

B24CS3T03	COMPILER DESIGN	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course introduces principles and techniques used in the design and implementation of compilers. It covers all phases from lexical analysis to code generation and optimization, focusing on algorithms, grammar models, syntax-directed translation, and symbol management. Students learn to build efficient compilers and interpret program semantics.

Prerequisites

Formal Languages and Automata Theory, Data Structures, and Computer Organization.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the phases of a compiler and apply lexical analysis techniques such as token specification and recognition using input buffering (Cognitive Knowledge Level: Apply)
CO 2	Apply context-free grammar transformations and construct top-down parsers including recursive-descent and LL(1) predictive parsers (Cognitive Knowledge Level: Apply)
CO 3	Demonstrate bottom-up parsing techniques by performing handle pruning, shift-reduce parsing, and constructing SLR, LR(1), and LALR parsing tables (Cognitive Knowledge Level: Apply)
CO 4	Develop syntax-directed definitions and generate intermediate code using three-address representations such as quadruples and triples. (Cognitive Knowledge Level: Apply)
CO 5	Develop and analyze sustainable code optimization techniques and design code generation strategies for a target machine. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1	2	2	2	1					2
CO 2	2	1	2	2	1	1					2
CO 3	2	1	2	1	2	1					2
CO 4	2	1	2	1	1	1					2
CO 5	2	2	2	1		2	1				2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignments)	5 marks
Assignment/Analytical	

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

INTRODUCTION TO COMPILER AND LEXICAL ANALYSIS

Analysis of the source program - Analysis and synthesis phases, Phases of a compiler. Compiler writing tools. Bootstrapping. Lexical Analysis - Role of Lexical Analyser, Input Buffering, Specification of Tokens, Recognition of Tokens.

MODULE 2 (9 hours)

INTRODUCTION TO SYNTAX ANALYSIS

Role of the Syntax Analyser – Syntax error handling. Review of Context Free Grammars - Derivation and Parse Trees, Eliminating Ambiguity. Basic parsing approaches - Eliminating left recursion, left factoring. Top-Down Parsing - Recursive Descent parsing, Predictive Parsing- LL(1) Grammars.

MODULE 3 (9 hours)

BOTTOM-UP-PARSING

Handle Pruning. Shift Reduce parsing. Operator precedence parsing . LR parsing - Constructing SLR, LALR and canonical LR parsing tables.

MODULE 4 (9 hours)

SYNTAX DIRECTED TRANSLATION AND INTERMEDIATE CODE GENERATION

Syntax directed translation - Syntax directed definitions, S-attributed definitions, L-attributed definitions, Type system and Type checking. Intermediate Code Generation - Intermediate languages, Graphical representations, Three-Address code, Quadruples, Triples.

MODULE 5 (9 hours)

CODE OPTIMIZATION AND GENERATION

Code Optimization– Introduction, Opportunities for Optimization, Scope of Optimization. Local Optimization: Local Value Numbering, Tree-Height Balancing. Regional Optimization: Superlocal Value Numbering, Loop Unrolling. Global Optimization: Finding Uninitialized Variables with Live Sets, Global Code Placement. Code generation -

Issues in the design of a code generator, Target Language, A simple code generator.

Text Books

1. Aho, Sethi & Ullman, Compilers: Principles, Techniques and Tools, Pearson

Reference Books

1. D.M.Dhamdhare, System Programming and Operating Systems, Tata McGraw Hill & Company, 1996.
2. Keith D. Cooper, Linda Torczon, Engineering a Compiler, Elsevier Science 3/e, 2023
3. Kenneth C. Loudon, Compiler Construction – Principles and Practice, Cengage Learning Indian Edition, 2006
4. Tremblay and Sorenson, The Theory and Practice of Compiler Writing, Tata McGraw Hill & Company, 1984.
5. John Levine, Flex & Bison: Text Processing Tools, O'Reilly Media, 2009.
6. Keith D. Cooper and Linda Torczon, Engineering a Compiler, 2nd Edition, Morgan Kaufmann, 2011.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No.of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction To Compiler And Lexical Analysis		9 hours
1.1	Introduction to compilers, Analysis of the source program	1
1.2	Phases of the compiler – Analysis Phases	1
1.3	Phases of the Compiler - Synthesis Phases	2
1.4	Symbol Table Manager and Error Handler	1
1.5	Compiler writing tools, bootstrapping	1
1.6	The role of Lexical Analyzer , Input Buffering	1
1.7	Specification of Tokens	1
1.8	Recognition of Tokens	1

Module 2: Introduction to Syntax Analysis		9 hours
2.1	Role of the Syntax Analyser, Syntax error handling	1
2.2	Review of Context Free Grammars	1
2.3	Parse Trees and Derivations	1
2.4	Grammar transformations, Eliminating ambiguity	1
2.5	Eliminating left recursion	1
2.6	Left factoring the grammar	1
2.7	Recursive Descent parsing	1
2.8	First and Follow	1
2.9	Predictive Parsing table constructor-LL(1) Grammars	1
Module 3: Bottom-Up-Parsing		9 hours
3.1	Bottom-up parsing - Handle Pruning	1
3.2	Shift Reduce parsing	1
3.3	Operator precedence parsing (Concept only)	1
3.4	LR parsing , SLR Grammar, items	1
3.5	Augmented Grammar, Canonical collection of LR(0) items	1
3.6	SLR Parser Table Construction	1
3.7	Constructing Canonical LR Parsing Tables	1
3.8	Constructing LALR Parsing Tables	1
3.9	LALR parser	1
Module 4: Syntax Directed Translation & Intermediate Code Generation		9 hours
4.1	Introduction to Syntax Directed Translation	1
4.2	Syntax Directed Definitions (SDD)	1
4.3	S-Attributed Definitions	1
4.4	L-Attributed Definitions	1
4.5	Type System: Type Expressions	1
4.6	Type Checking (Static and Dynamic)	1
4.7	Intermediate Languages and Graphical Representations (DAGs)	1
4.8	Three-Address Code — Types and Representation	1
4.9	Quadruples, Triples and Translation Examples	1

Module 5: Code Optimization and Generation		9 hours
5.1	Introduction to Code Optimization, Need and Categories	1
5.2	Opportunities and Scope of Optimization	1
5.3	Local Optimization: Local Value Numbering	1
5.4	Local Optimization: Tree-Height Balancing	1
5.5	Regional Optimization: Superlocal Value Numbering	1
5.6	Regional Optimization: Loop Unrolling	1
5.7	Global Optimization: Live Sets and Finding Uninitialized Variables	1
5.8	Global Code Placement	1
5.9	Code Generation – Issues, Target Language, Simple Code Generator	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the phases of a compiler with a neat diagram showing information flow between each phase.
2. Describe how lexical analyzers use finite automata for token recognition

Course Outcome 2 (CO 2):

1. Construct the FIRST and FOLLOW sets for a given grammar and show how an LL(1) parse table is built.
2. Compare top-down and bottom-up parsing strategies with examples.

Course Outcome 3 (CO 3):

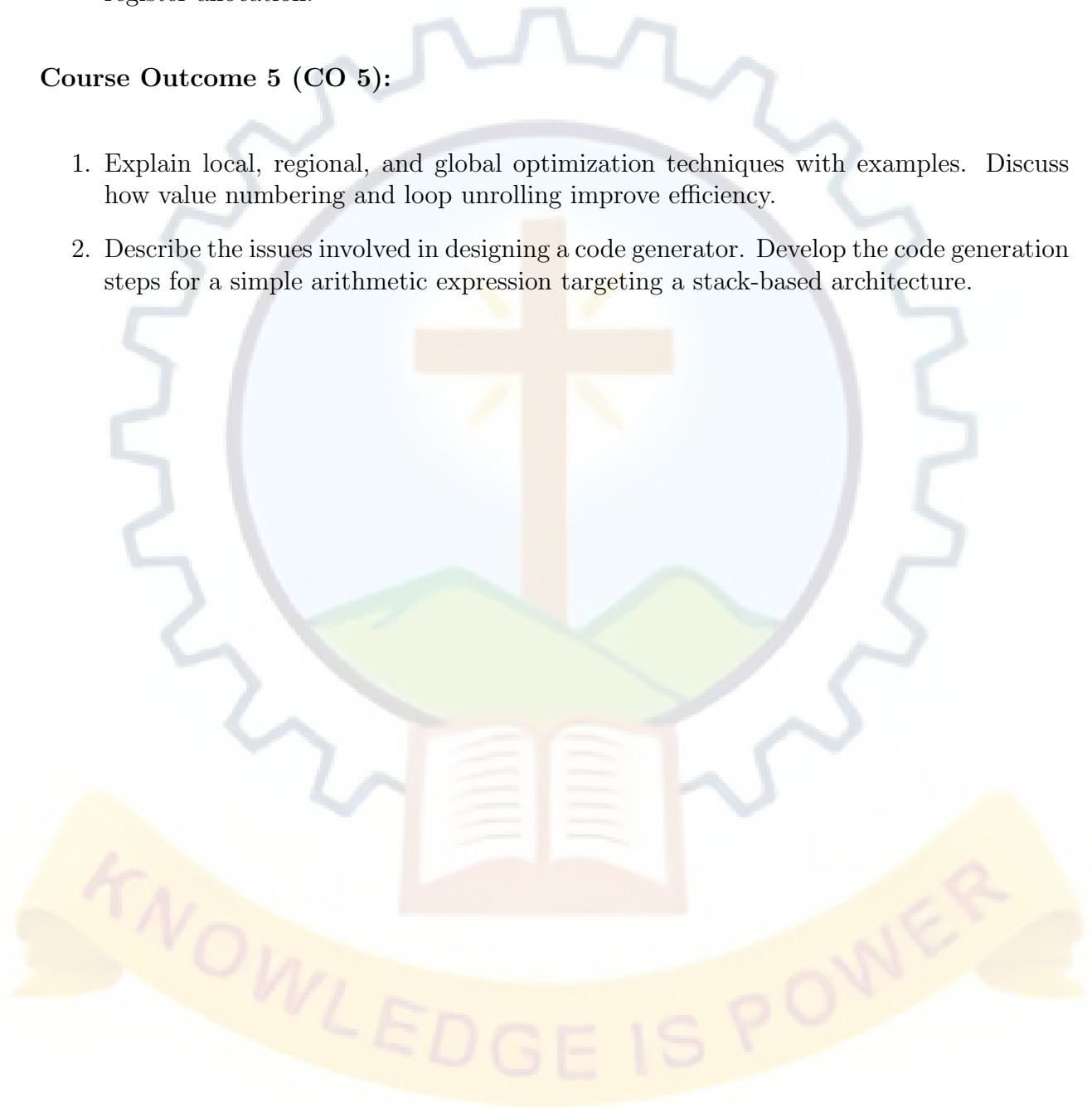
1. Develop syntax-directed definitions to generate postfix expressions for the arithmetic grammar.
2. Explain activation records and storage organization with a call chain example.

Course Outcome 4 (CO 4):

1. Illustrate common code optimization techniques used in compiler design with examples.
2. Design a simple code generation algorithm for three-address statements and explain register allocation.

Course Outcome 5 (CO 5):

1. Explain local, regional, and global optimization techniques with examples. Discuss how value numbering and loop unrolling improve efficiency.
2. Describe the issues involved in designing a code generator. Develop the code generation steps for a simple arithmetic expression targeting a stack-based architecture.



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CS3T03

Course Name: COMPILER DESIGN

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Distinguish between compiler and interpreter..
2. Explain the role of lexical analyzer in compiler design.
3. Differentiate between top-down and bottom-up parsing.
4. Write the advantages of predictive parsing over recursive descent parsing.
5. Define handle and handle pruning in bottom-up parsing.
6. Differentiate between S-attributed and L-attributed definitions with an example.
7. Define quadruples and triples with examples.
8. What is peephole optimization? Give an example.
9. What is local value numbering? Explain briefly with a small example.
10. List any three issues in the design of a code generator and explain why they are important.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) With a neat diagram, explain the phases of a compiler. (8)
(b) Discuss bootstrapping and its importance in compiler construction. (6)

OR

12. (a) Explain lexical analysis process and input buffering techniques.. (8)
(b) Design a finite automaton for token recognition of identifiers. (6)
13. (a) Eliminate left recursion and left factor the following grammar (8)
(b) Explain LL(1) parsing with parse table construction. (6)

OR

14. (a) Describe recursive-descent parsing method with an example. . (8)
(b) Discuss error handling strategies in syntax analysis. (6)
15. (a) Explain shift-reduce parsing with an example (8)
(b) Construct an SLR parse table for a given grammar (6)

OR

16. (a) Compare SLR, LALR and Canonical LR parsers. (8)
(b) Explain the working of YACC parser generator (6)
17. (a) Explain the construction of three-address code from a syntax-directed definition. (8)
(b) Describe storage organization and activation records in a compiler. (6)

OR

18. (a) Explain translation schemes for arithmetic expressions with syntax trees. (8)
(b) Compare quadruples and triples representations with examples. . (6)
19. (a) Describe the process of global code placement and finding uninitialized variables using live sets. Provide an example. (8)
(b) What are the major issues in the design of a code generator? Illustrate how a simple code generator works using a stack-based target machine. (6)

OR

20. (a) Differentiate machine-dependent and machine-independent optimizations with examples. (8)
(b) Explain a simple algorithm for code generation from three-address code. (6)

B24CS3T04	ALGORITHM ANALYSIS AND DESIGN	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

The course introduces students to the design of computer algorithms, as well as analysis of algorithms. Algorithm design and analysis provide the theoretical backbone of computer science and are a must in the daily work of the successful programmer. The goal of this course is to provide a solid background in the design and analysis of the major classes of algorithms. At the end of the course students will be able to develop their own versions for a given computational task and to compare and contrast their performance.

Prerequisite

Strong Foundation in Mathematics, Programming in C, Data Structures and Graph Theory.

Course Outcomes

After the completion of the course the student will be able to:

CO1	Analyze any given algorithm and express its time and space complexities in asymptotic notations. (Cognitive Knowledge Level: Apply)
CO2	Derive recurrence equations and solve it using Iteration, Recurrence Tree, Substitution and Master's Method to compute time complexity of algorithms. (Cognitive Knowledge Level: Apply)
CO3	Illustrate Graph traversal algorithms & applications and Advanced Data structures like AVL trees and Disjoint set operations. (Cognitive Knowledge Level: Apply)
CO4	Demonstrate Divide-and-conquer, Greedy Strategy, Dynamic programming, Branch-and-Bound and Backtracking algorithm design techniques. (Cognitive Knowledge Level: Apply)
CO5	Classify a problem as computationally tractable or intractable, and discuss strategies to address intractability and Identify the suitable design strategy to solve a given problem. (Cognitive Knowledge Level: Analyze)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	2	3	2	1			1	2	1
CO2	2	1		2	1				1		
CO3	2	1	2	1	1			1	1		1
CO4	2	2	3	2	2	1		2	3	3	2
CO5	1	2	1	2	1	3	2	1	1	1	2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contains 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE-1 (9 hours)

Introduction to Algorithm Analysis:

Characteristics of Algorithms, Criteria for Analysing Algorithms, Time and Space Complexity - Best, Worst and Average Case Complexities, Asymptotic Notations - Big-Oh (O), Big- Omega (Ω), Big-Theta (θ), Little-oh (o) and Little- Omega (ω) and their properties. Classifying functions by their asymptotic growth rate, Time and Space Complexity Calculation of simple algorithms.

Analysis of Recursive Algorithms: Recurrence Equations, Solving Recurrence Equations – Iteration Method, Recursion Tree Method, Substitution method and Master's Theorem (Proof not required).

MODULE-2 (10 hours)

Advanced Data Structures and Graph Algorithms:

Self Balancing Tree - AVL Trees (Insertion and deletion operations with all rotations in detail, algorithms not expected); Disjoint Sets-Disjoint set operations, Union and find algorithms.

DFS and BFS traversals - Analysis, Strongly Connected Components of a Directed graph, Topological Sorting.

MODULE-3 (8 hours)

Divide & Conquer and Greedy Strategy:

The Control Abstraction of Divide and Conquer- 2-way Merge sort, Strassen's Algorithm for Matrix Multiplication-Analysis. The Control Abstraction of Greedy Strategy- Fractional Knapsack Problem, Minimum Cost Spanning Tree Computation- Kruskal's Algorithms, Prim's algorithm - Analysis, Single Source Shortest Path Algorithm - Dijkstra's Algorithm-Analysis.

MODULE-4 (8 hours)

Dynamic Programming, Back Tracking and Branch & Bound:

The Control Abstraction- The Optimality Principle- Matrix Chain Multiplication-Analysis, All Pairs Shortest Path Algorithm - Floyd-Warshall Algorithm-Analysis. The Control Abstraction of Back Tracking – The N Queen's Problem. Branch and Bound Algorithm

for Travelling Salesman Problem.

MODULE-5 (10 hours)

Introduction to Complexity Theory:

Tractable and Intractable Problems, Complexity Classes – P, NP, NP- Hard and NP-Complete Classes- NP Completeness proof of Clique Problem and Vertex Cover Problem- Approximation algorithms- Bin Packing, Graph Coloring. Randomized Algorithms (Definitions of Monte Carlo and Las Vegas algorithms), Randomized version of Quick Sort algorithm with analysis.

Text Books

1. T.H.Cormen, C.E.Leiserson, R.L.Rivest, C. Stein, Introduction to Algorithms, 2nd Edition, Prentice-Hall India (2001)
2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, “Fundamentals of Computer Algorithms”, 2nd Edition, Orient Longman Universities Press (2008)
3. Sara Baase and Allen Van Gelder, Computer Algorithms, Introduction to Design and Analysis, 3rd Edition, Pearson Education (2009)

Reference Books

1. Jon Kleinberg, Eva Tardos, “Algorithm Design”, First Edition, Pearson (2005)
2. Robert Sedgewick, Kevin Wayne, “Algorithms”, 4th Edition Pearson (2011)
3. Gilles Brassard, Paul Bratley, “Fundamentals of Algorithmics”, Pearson (1996)
4. Steven S. Skiena, “The Algorithm Design Manual”, 2nd Edition, Springer (2008)

COURSE CONTENTS AND TEACHING PLAN (45 hrs)

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction to Algorithm Analysis		9 hours
1.1	Introduction to Algorithm Analysis: Characteristics of Algorithms.	1
1.2	Criteria for Analysing Algorithms, Time and Space Complexity - Best, Worst and Average Case Complexities.	1

1.3	Asymptotic Notations - Properties of Big-Oh (O), Big-Omega (Ω), Big-Theta (θ), Little-Oh (o) and Little- Omega (ω).	1
1.4	Illustration of Asymptotic Notations.	1
1.5	Classifying functions by their asymptotic growth rate.	1
1.6	Time and Space Complexity Calculation of algorithms/code segments.	1
1.7	Analysis of Recursive Algorithms: Recurrence Equations, Solving Recurrence Equations – Iteration Method.	1
1.8	Recursion Tree Method.	1
1.9	Substitution method and Master's Theorem and its Illustration.	1
Module 2: Advanced Data Structures and Graph Algorithms		10 hours
2.1	Self Balancing Trees - Properties of AVL Trees, Rotations of AVL Trees.	1
2.2	AVL Trees Insertion and Illustration.	1
2.3	AVL Trees Deletion and Illustration.	1
2.4	Disjoint set operations.	1
2.5	Union and find algorithms.	1
2.6	Illustration of Union and find algorithms.	1
2.7	Graph Algorithms: BFS traversal, Analysis.	1
2.8	DFS traversal, Analysis.	1
2.9	Strongly connected components of a Directed graph.	1
2.10	Topological Sorting.	1
Module 3: Divide & Conquer and Greedy Method		8 hours
3.1	Divide and Conquer: The Control Abstraction.	1
3.2	2-way Merge Sort, Analysis.	1
3.3	Strassen's Algorithm for Matrix Multiplication, Analysis.	1
3.4	Greedy Strategy: The Control Abstraction.	1
3.5	Fractional Knapsack Problem.	1
3.6	Minimum Cost Spanning Tree Computation- Kruskal's Algorithm, Prim's algorithm, Analysis.	1

3.7	Single Source Shortest Path Algorithm - Dijkstra's Algorithm.	1
3.8	Illustration of Dijkstra's Algorithm-Analysis.	1
Module 4: Dynamic Programming, Back Tracking and Branch and Bound		8 hours
4.1	Dynamic Programming: The Control Abstraction, The Optimality Principle.	1
4.2	Matrix Chain Multiplication-Analysis.	1
4.3	Illustration of Matrix Chain Multiplication-Analysis.	1
4.4	All Pairs Shortest Path Algorithm- Analysis and Illustration of Floyd-Warshall Algorithm.	1
4.5	Back Tracking: The Control Abstraction.	1
4.6	Back Tracking: The N Queen's Problem.	1
4.7	Branch and Bound:- Travelling salesman problem.	1
4.8	Branch and Bound:- Travelling salesman problem.	1
Module 5: Introduction to Complexity Theory		10 hours
5.1	Introduction to Complexity Theory: Tractable and Intractable Problems.	1
5.2	Complexity Classes - P, NP.	1
5.3	NP- Hard and NP-Complete Problems.	1
5.4	NP Completeness Proof of Clique Problem.	1
5.5	NP Completeness Proof of Vertex Cover Problem.	1
5.6	Approximation algorithms- Bin Packing Algorithm and Illustration.	1
5.7	Graph Colouring Algorithm and Illustration.	1
5.8	Randomized Algorithms (definitions of Monte Carlo and Las Vegas algorithms).	1
5.9	Randomized Version of Quick Sort Algorithm with Analysis.	1
5.10	Illustration of Randomized Version of Quick Sort Algorithm with Analysis.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

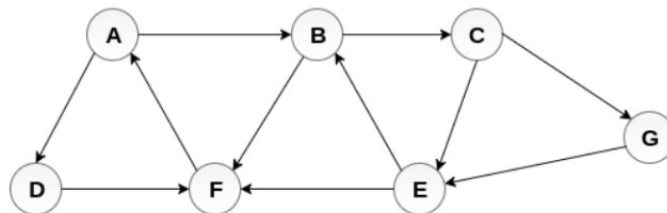
1. Is $2n + 1 = O(2^n)$? Is $2^{2n} = O(2^n)$? Justify your answer.
2. What is the need of asymptotic analysis in calculating time complexity? What are the notations used for asymptotic analysis?
3. Calculate the time complexity for addition of two matrices.
4. Define time complexity and space complexity. Write an algorithm for adding n natural numbers and analyse the time and space requirements of the algorithm.

Course Outcome 2 (CO2):

1. State Master's theorem for solving recurrences.
2. Solve the recurrence $T(n) = 3T(n - 2)$, using iteration method.
3. State the conditions in recurrences where Master Theorem is not applicable.
4. Solve the following recurrence equations using Master's theorem:
 - (a) $T(n) = 8T(n/2) + 100n^2$
 - (b) $T(n) = 2T(n/2) + 10n$
5. Using Recursion Tree method, Solve $T(n) = 2T(n/10) + T(9n/10) + n$. Assume constant time for small values of n .

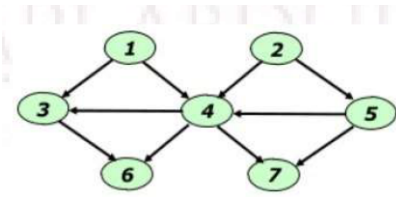
Course Outcome 3 (CO3):

1. Explain the rotations performed for insertion in AVL tree with example.
2. Write down BFS algorithm and analyse the time complexity. Perform BFS traversal on the given graph starting from node A. If multiple node choices are available for next travel, choose the next node in alphabetical order.



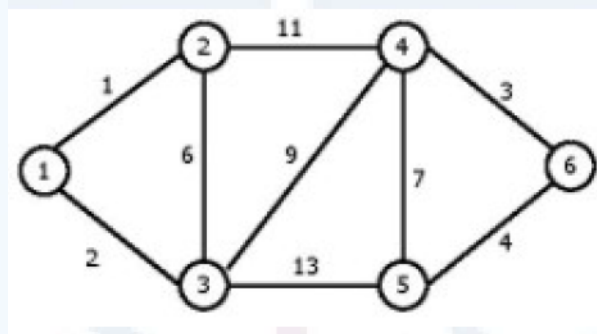
3. Find the minimum and maximum height of any AVL-tree with 7 nodes? Assume that the height of a tree with a single node is 0.

- Find any three topological orderings of the given graph.

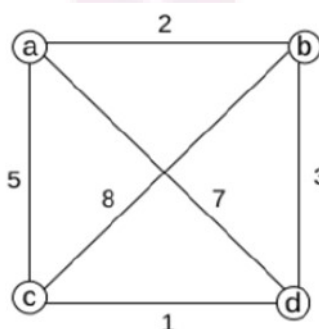


Course Outcome 4 (CO4):

- Give the control abstraction for Divide and Conquer method.
- Construct the minimum spanning tree for the given graph using Kruskal's algorithm. Analyse the complexity of the algorithm.



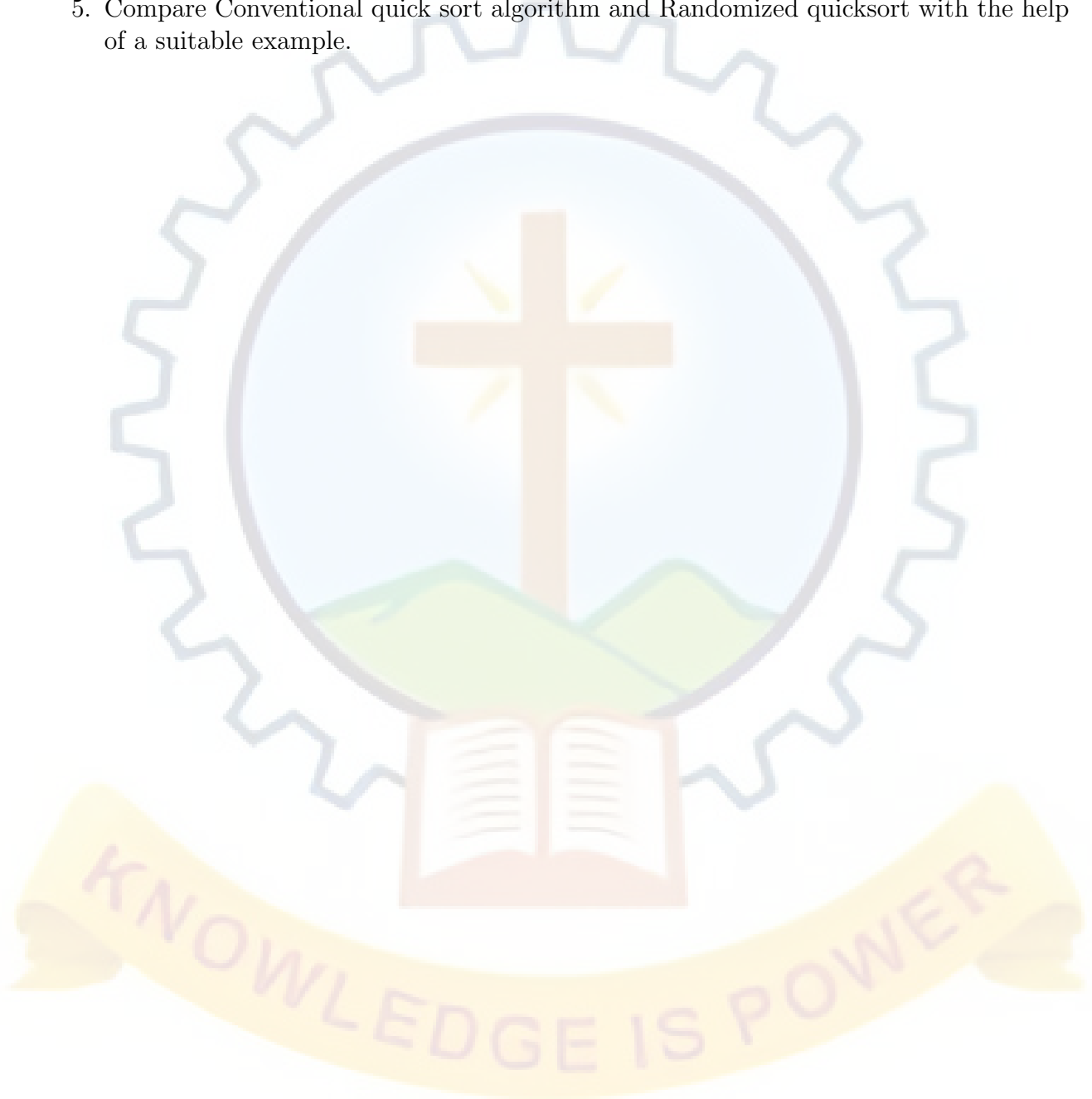
- Compare Divide and Conquer and Dynamic programming methodologies.
- What is Principle of Optimality?
- Define Travelling Salesman Problem (TSP). Apply branch and bound algorithm to solve TSP for the given graph, assuming the start city as 'a'. Draw the state space tree.



Course Outcome 5 (CO5):

- Compare Tractable and Intractable Problems.

2. With the help of suitable code sequence convince Vertex Cover Problem is an example of NP-Complete Problem.
3. Explain Vertex Cover problem using an example. Suggest an algorithm for finding Vertex Cover of a graph.
4. Write short notes on approximation algorithms.
5. Compare Conventional quick sort algorithm and Randomized quicksort with the help of a suitable example.



MODEL QUESTION PAPER

QP CODE:

Pages: 4

Reg.No.:

Name:

MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CS3T04

Course Name: Algorithm Analysis and Design

Max. Marks: 100

Duration: 3 Hours

PART A

Answer All Questions. Each Question Carries 3 Marks.

1. Define asymptotic notation? Arrange the following functions in increasing order of asymptotic growth rate.

$n^3, 2^n, \log n^3, 2^{100}, n^2 \log n, n^n, \log n, n^{0.3}, 2^{\log n}$

2. State Master's Theorem. Find the solution to the following recurrence equations using Master's theorem.

(a) $T(n) = 8T(n/2) + 100n^2$

(b) $T(n) = 2T(n/2) + 10n$

3. Find any two topological ordering of the DAG given below.
4. Show the UNION operation using linked list representation of disjoint sets.
5. Write the control abstraction of greedy strategy to solve a problem.
6. Write an algorithm based on divide-and-conquer strategy to search an element in a given list. Assume that the elements of list are in sorted order.
7. List the sequence of steps to be followed in Dynamic Programming approach.
8. Illustrate how optimal substructure property could be maintained in Floyd-Warshall algorithm.

9. Differentiate between P and NP problems.
10. Specify the relevance of approximation algorithms.

PART B

Answer any one question from each module. Each question carries 14 Marks.

11. (a) Define Big O, Big Ω and Big Θ Notation and illustrate them graphically. (7)
- (b) Solve the following recurrence equation using recursion tree method

$$T(n) = T(n/3) + T(2n/3) + n, \text{ where } n > 1; \quad T(n) = 1, \text{ otherwise.} \quad (7)$$

OR

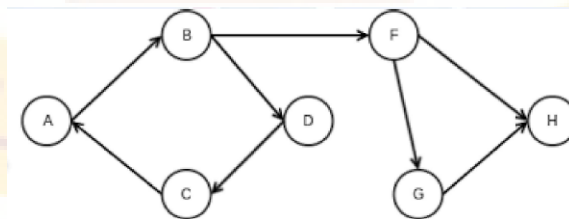
12. (a) Explain the iteration method for solving recurrences and solve the following recurrence equation using iteration method.

$$T(n) = 3T(n/3) + n; \quad T(1) = 1 \quad (7)$$

- (b) Determine the time complexities of the following two functions fun1() and fun2().

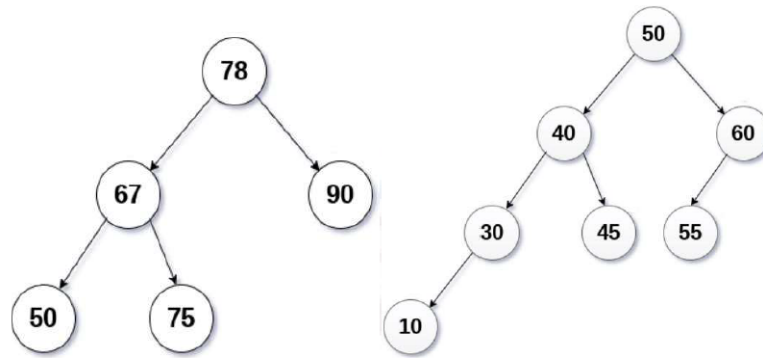
```
int fun1(int n){ if (n != 1) return n; return 2*fun1(n-1);}
int fun2 (int n){ if (n != 1) return n; return fun2 (n-1) + fun2 (n-1);} \quad (7)
```

13. (a) Write DFS algorithm and analyse its time complexity. Illustrate the classification of edges in DFS traversal. (7)
- (b) Find the strongly connected components of the digraph given below: (7)



OR

14. (a) Illustrate the advantage of height balanced binary search trees over binary search trees? Explain various rotations in AVL trees with example. (7)
- (b) Perform the following operations in the given AVL trees. i) Insert 70 ii) Delete 55 (7)



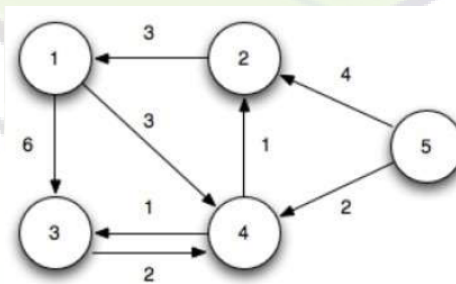
15. (a) State Fractional Knapsack Problem and write Greedy Algorithm for Fractional Knapsack Problem. (7)
- (b) Find the optimal solution for the following Fractional Knapsack problem. Given the number of items(n) = 7, capacity of sack(m) = 15,

$$W = \{2, 3, 5, 7, 1, 4, 1\} \text{ and } P = \{10, 5, 15, 7, 6, 18, 3\}$$

(7)

OR

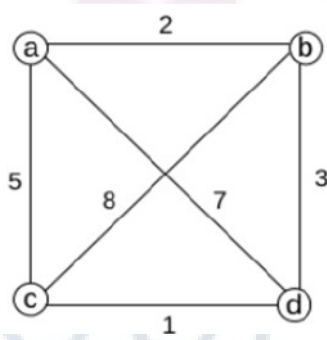
16. (a) Write and explain merge sort algorithm using divide and conquer strategy using the data $\{30, 19, 35, 3, 9, 46, 10\}$. Also analyse the time complexity. (7)
- (b) Write the pseudo code for Dijkstra's algorithm. Compute the shortest distance from vertex 1 to all other vertices of the given graph using Dijkstra's algorithm. (7)



17. (a) Write Floyd-Warshall algorithm and analyse its complexity. (5)
- (b) Write and explain the algorithm to find the optimal parenthesization of matrix chain product whose sequence of dimension is $4 \times 10, 10 \times 3, 3 \times 12, 12 \times 20$. (9)

OR

18. (a) Explain the concept of Backtracking method using 4 Queens problem. (7)
- (b) Define Travelling Salesman Problem (TSP). Apply branch and bound algorithm to solve TSP for the following graph, assuming the start city as 'a'. Draw the state space tree. (7)



19. (a) State bin packing problem? Explain the first fit decreasing strategy (7)
(b) Prove that the Clique problem is NP-Complete. (7)

OR

20. (a) Explain the need for randomized algorithms. Differentiate Las Vegas and Monte Carlo algorithms. (6)
(b) Explain randomized quicksort and analyse the expected running time of randomized quicksort with the help of a suitable example? (9)

B24CS3T05	ARTIFICIAL INTELLIGENCE	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course provides a foundation in the principles and techniques of Artificial Intelligence, emphasizing how intelligent agents perceive, reason, and act in different environments. It covers classical search methods, game-playing strategies, constraint satisfaction techniques, and logical knowledge representation using propositional and First-Order Logic. By completing this course, students gain the ability to model, analyze, and design intelligent systems capable of solving complex problems.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Describe the foundations, history, applications, and agent-based models of Artificial Intelligence. (Cognitive Knowledge Level: Understanding)
CO 2	Apply uninformed and informed search strategies to solve well-defined AI problems. (Cognitive Knowledge Level: Apply)
CO 3	Analyze and apply local search, game-playing algorithms, and CSP techniques to complex AI problem environments.(Cognitive Knowledge Level: Apply)
CO 4	Demonstrate knowledge representation using propositional logic and First-Order Logic (FOL), and perform logical inference using resolution, unification, and forward/backward chaining.(Cognitive Knowledge Level: Apply)
CO 5	Describe modern AI approaches, machine learning paradigms, their integration with search, and ethical implications with case studies. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	2	1	1		2			1		2
CO 2	3	3	2	3	3	1			1	1	2
CO 3	3	3	2	3	3	1			1	2	2
CO 4	3	3	2	3	3				1		2
CO 5	2	3	3	3	2	3	3	3	3	2	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	15	15	15
Understand	40	40	40
Apply	45	45	45
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/video assignments/analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction & Intelligent Agents

Introduction-What is Artificial Intelligence, History of AI, Applications of AI, and Foundations of AI, Agenting in AI, Responsible AI, Intelligent agents-Agents and its Environments, Good Behavior- The concept of rationality, Nature of Environments -Specifying the task environment, Properties of task environment, Environment types-observable, deterministic, episodic, static, discrete Agents and environments, Agent architectures-Simple reflex, Model-based, Goal-based, Utility-based Agents.

MODULE 2 (6 hours)

Problem Solving By Searching

Problem formulation, state space, Trade-offs in BFS & DFS, Uninformed search: Depth Limited Search, Iterative Deepening Depth Limited Search- performance measures. Informed Search: Heuristic functions, Greedy best-first, A* - admissibility & consistency.

MODULE 3 (9 hours)

Search in Complex Environments

Adversarial Search Game playing: Minimax, Alpha-Beta pruning, Evaluation functions, Constraint Satisfaction Problems (CSP): Defining CSP -variables, domains, constraints, Example Problems- Crypt arithmetic Problem, Map Coloring problem, Backtracking-MRV/degree/LCV, forward checking, Constraint Propagation -Inference in CSP, Local Search Strategies-Simple Hill Climbing, Simulated Annealing.

MODULE 4 (7 hours)

Knowledge Representation and Reasoning

Knowledge-Based Agents: The Wumpus world. Logic and Propositional Logic: Syntax and Semantics, Entailment, Inference, Resolution. First-Order Logic (FOL): Syntax and Semantics of FOL, Knowledge Representation in FOL. Inference in FOL: Unification and Lifting, Forward/Backward chaining, Resolution. Algorithms for Planning - State Space Search, Planning Graphs.

MODULE 5 (7 hours)

Towards Modern AI- Bridging Symbolic AI & Machine Learning

The Evolution to Data-Driven AI: Core limitations of symbolic approaches. Machine Learning paradigms: Supervised, Unsupervised, Reinforcement Learning. Integration with

Core AI: How search informs Reinforcement Learning. Modern game playing (AlphaGo) combines search with learned models. Ethics & Responsibility: Bias, fairness, and societal impact of learned models. Case Study: Healthcare, Autonomous Systems, and Intelligent Assistants.

Text Books

1. S. Russell, P. Norvig P, Artificial Intelligence: A Modern Approach, 3e/4e, Pearson Prentice Hall(2014).
2. D. Poole, A. Mackworth, Artificial Intelligence: Foundations of Computational Agents, 2e, Cambridge

Reference Books

1. Rich E., Knight K. and Nair B. S., Artificial Intelligence, Tata McGraw Hills (2009) 3e
2. Luger F. G., Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education Asia (2009) 6e.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction Intelligent Agents		7 hours
1.1	Introduction: What is AI, History, Foundations and Applications.	2
1.2	Intelligent Agents, Agenting in AI and Responsible AI	1
1.3	Concept of Rationality & Nature of Environments	1
1.4	Task Environment Properties & Environment Types	1
1.5	Agent Architectures: Reflex, Model-Based, Goal-Based, Utility-Based Agents	1
1.6	Goal-Based, Utility-Based Agent	1
Module 2: Problem Solving by Searching		6 hours
2.1	Problem formulation and state space	1
2.2	Uninformed Search – BFS, DFS	1

2.3	Depth First Search and Depth Limited Search	1
2.4	Iterative Deepening Search and performance measures	1
2.5	Heuristics and Greedy Best-first	1
2.6	A* Search and Admissibility and Consistency	1
Module 3: SSearch in Complex Environments		9 hours
3.1	Game Playing – Minimax	1
3.2	Alpha–Beta Pruning	1
3.3	Evaluation Functions, Cut-off, Move Ordering	1
3.4	CSP Basics: Variables, Domains, Constraints	1
3.5	Example CSPs – Cryptarithmic, Map Coloring	1
3.6	Backtracking Search	1
3.7	MRV, Degree Heuristic, LCV, Forward Checking	1
3.8	Constraint Propagation	1
3.9	Local Search: Hill Climbing & Simulated Annealing (Idea)	1
Module 4: Knowledge Representation and Reasoning		7 hours
4.1	Knowledge-Based Agents and the Wumpus World	2
4.2	Propositional Logic: Syntax and Semantics	1
4.3	Entailment and Inference in Propositional Logic	1
4.4	Entailment and Inference in Propositional Logic	1
4.5	First-Order Logic (FOL): Syntax, Semantics & Knowledge Representation	1
4.6	Inference in First-Order Logic: Unification, Forward & Backward Chaining	1
4.7	Planning Algorithms: State Space Search and Planning Graphs	1
Module 5: Towards Modern AI- Bridging Symbolic AI & Machine Learning		7 hours
5.1	Evolution to Data-Driven AI and Limitations of Symbolic Approaches	1
5.2	Machine Learning Paradigms: Supervised Learning	1
5.3	Machine Learning Paradigms: Unsupervised Learning	1
5.4	Machine Learning Paradigms: Reinforcement Learning & Lifting	1

5.5	Integration of Core AI with Learning	1
5.6	Ethics and Responsibility in AI: Bias, Fairness, and Societal Impact	1
5.7	Case Studies: Healthcare, Autonomous Systems, and Intelligent Assistants	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the properties of different task environments with suitable examples. How do these properties influence the design of intelligent agents?
2. Formulate any real-life problem (e.g., route planning, robot navigation, or scheduling) as an AI problem by clearly stating the initial state, actions, transition model, goal state, and performance measure.

Course Outcome 2 (CO 2):

1. Write the difference between DFS and DLS.
2. What is an admissible heuristic?

Course Outcome 3 (CO 3):

1. Solve the given CSP (Cryptarithmic) using backtracking and apply MRV/Degree/LCV heuristics to reduce search effort.

$$\begin{array}{r} T W O \\ + T W O \\ \hline F O U R \end{array}$$

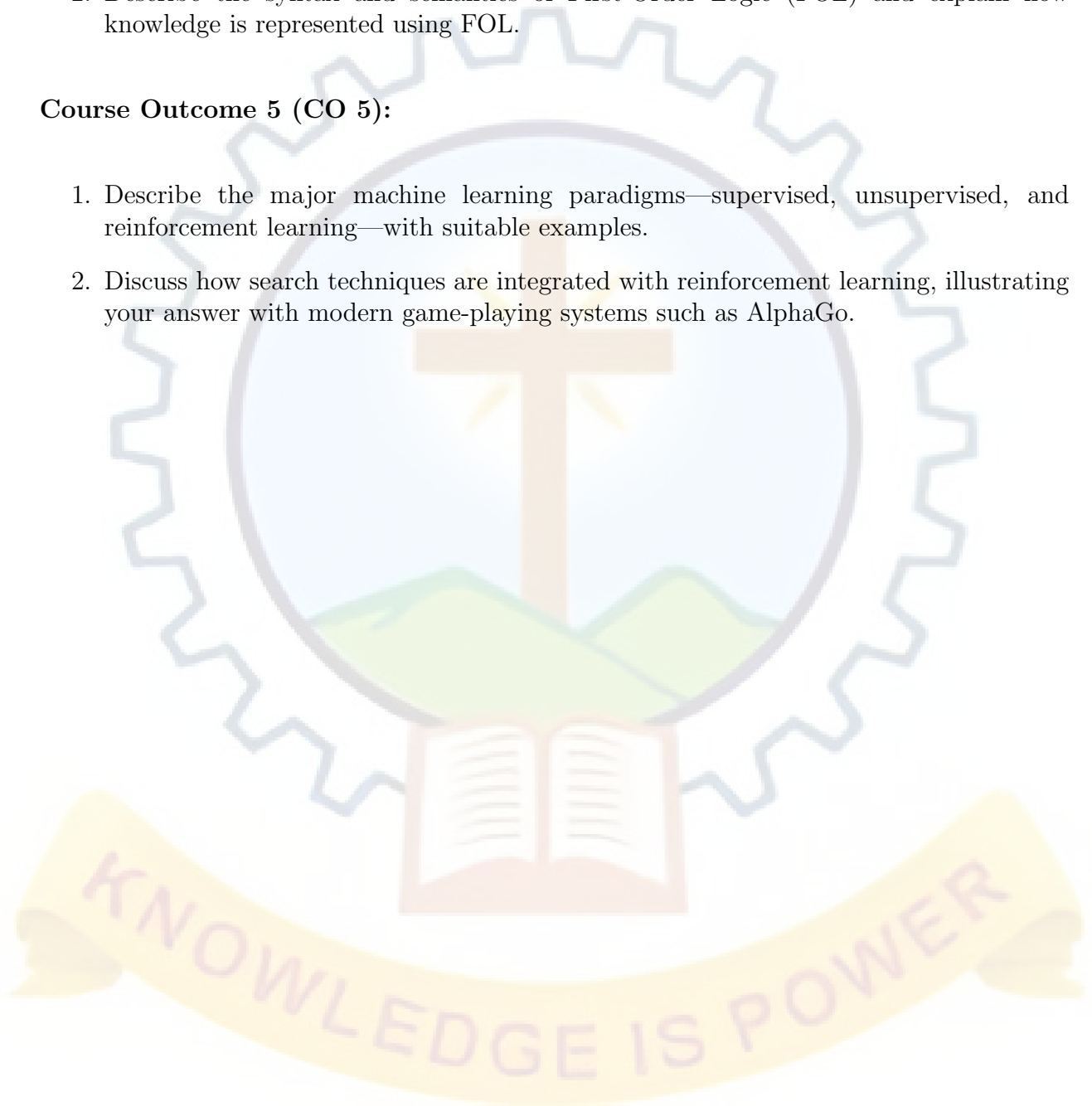
2. Explain the idea of simulated annealing.

Course Outcome 4 (CO 4):

1. Represent the following English statements in Propositional Logic, and use resolution to determine whether the conclusion logically follows. If it is raining then the ground is wet. It is raining. Conclusion to test: The ground is wet.
2. Describe the syntax and semantics of First-Order Logic (FOL) and explain how knowledge is represented using FOL.

Course Outcome 5 (CO 5):

1. Describe the major machine learning paradigms—supervised, unsupervised, and reinforcement learning—with suitable examples.
2. Discuss how search techniques are integrated with reinforcement learning, illustrating your answer with modern game-playing systems such as AlphaGo.



MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CS3T05

Course Name: ARTIFICIAL INTELLIGENCE

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define rationality in the context of intelligent agents.
2. Differentiate between static and dynamic environments with examples.
3. Write the differences between BFS and DFS in terms of completeness and optimality.
4. Define heuristic function. Give one example.
5. What is the purpose of an evaluation function in game playing?
6. Define MRV heuristic with an example.
7. Define entailment in propositional logic.
8. What is unification? Give a simple example ?
9. What are the core limitations of symbolic AI approaches that led to the evolution of data-driven AI?
10. Define Reinforcement Learning and mention its key components

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the properties of different task environments and discuss how each property influences the design of intelligent agents with examples. (7)
(b) Describe the PEAS representation for any two real-world intelligent agents. (7)

OR

12. (a) Explain the various agent architectures—simple reflex, model-based, goal-based, and utility-based—with suitable diagrams. (8)
(b) Discuss the foundations and history of Artificial Intelligence and explain how these shaped modern AI applications. (6)
13. (a) Explain the working of BFS, DFS, DLS, and IDS. Compare them using performance measures. (7)
(b) Describe Greedy Best-first Search and A* Search with examples. (7)

OR

14. (a) What are admissible and consistent heuristics? Explain with examples and show how they affect A*. (7)
(b) Explain IDS. Why is it better than BFS and DFS? (7)
15. (a) Discuss forward checking and constraint propagation. How do they improve CSP search efficiency? (7)
(b) Explain backtracking, MRV, Degree heuristic, and LCV in CSP search. (7)

OR

16. (a) What is a CSP? Explain CSP structure, constraints, and examples like Map Coloring & Cryptarithmic. (7)
(b) Discuss forward checking and constraint propagation. How do they improve CSP search efficiency? (7)
17. (a) Explain the syntax and semantics of propositional logic and show how inference is performed using resolution. (4)
(b) Describe the components of the Wumpus World environment. Why is it considered a good testbed for logic-based agents? (10)

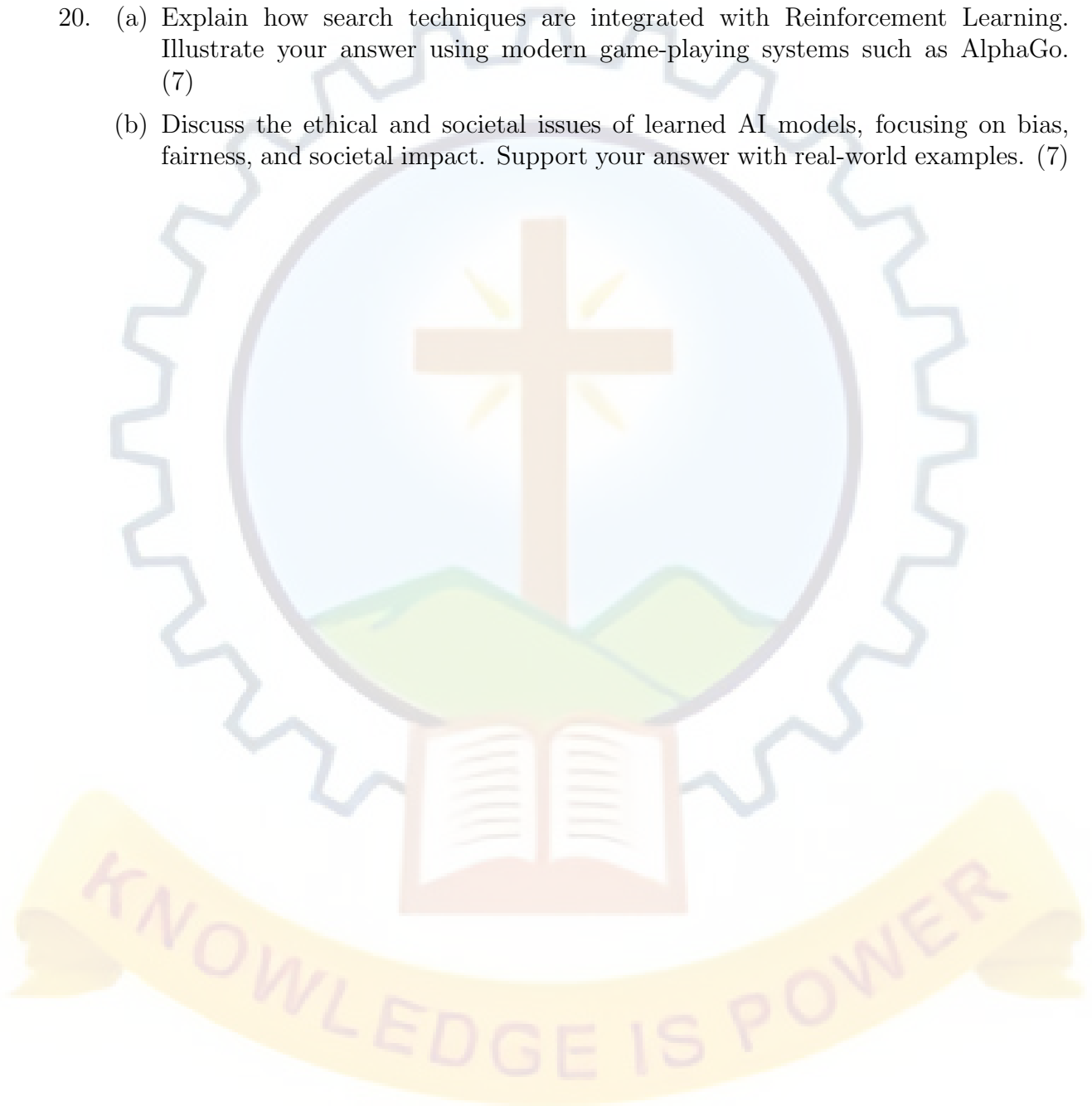
OR

18. (a) Discuss the process of Knowledge Representation in First-Order Logic (FOL). Explain how objects, relations, and quantifiers are represented using suitable examples. (8)
(b) What is Lifting in FOL? Explain how lifting is used to generalize resolution from propositional logic to first-order logic, with an example. (6)

19. (a) Explain the evolution from symbolic AI to data-driven AI. Discuss the limitations of symbolic approaches and how machine learning overcomes them. (8)
- (b) Describe the major machine learning paradigms—Supervised, Unsupervised, and Reinforcement Learning—with suitable examples. (6)

OR

20. (a) Explain how search techniques are integrated with Reinforcement Learning. Illustrate your answer using modern game-playing systems such as AlphaGo. (7)
- (b) Discuss the ethical and societal issues of learned AI models, focusing on bias, fairness, and societal impact. Support your answer with real-world examples. (7)



B24CS3P11	PROGRAMMING PARADIGMS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4		2024

Preamble

This course offers learners a solid foundation in the fundamental constructs of contemporary programming languages and the underlying concepts that influence their design. It helps the learners to equip with the knowledge necessary for the critical evaluation of existing and upcoming programming languages. It also enables the learner to choose the most appropriate language for a given programming task and to design new generation languages.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the criteria for evaluating programming languages and compare Imperative, Functional and Logic programming languages.(Cognitive Knowledge Level: Understand)
CO 2	Illustrate the characteristics of data types and variables. (Cognitive Knowledge Level: Apply)
CO 3	Comprehend how control flow structures and subprograms help in developing the structure of a program to solve a computational problem. (Cognitive Knowledge Level: Apply)
CO 4	Explain the characteristics of Object-Oriented Programming Languages. (Cognitive Knowledge Level: Understand)
CO 5	Compare concurrency constructs in different programming languages. (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	2	2	2				1		3
CO 2	2	2	1	2	2						
CO 3	2	3	2	3	3				1		
CO 4	2	2	2	2	3				1		3
CO 5	2	2	2	2	2				1		3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction to Programming Paradigms:

Introduction – Role of Programming Languages, Programming Domains, Language Evaluation Criteria, Influence on Language Design, Language Design Trade-offs, Implementation Methods. Names, Bindings and Scope – Names, Variables, Concept of Binding, Scope and Lifetime, Referencing Environments.

MODULE 2 (7 hours)

Data Types and Expressions in major programming languages:

Data Types – Primitive Data Types, Character String Types, User-Defined Ordinal Types, Array Types, Record Types, List Types, Pointer and Reference Types, Type Checking, Strong Typing, Type Equivalence. Expressions – Arithmetic Expressions, Overloaded Operators, Type Conversions, Relational and Boolean Expressions, Short-Circuit Evaluation. Assignment - Assignment Statements, Mixed-mode Assignment.

MODULE 3 (8 hours)

Flow Control and Subprogram Concepts in major programming languages:

Statement-Level Control Structures – Selection Statements, Iterative Statements, Unconditional Branching, Guarded Commands. Subprograms – Design Issues of Subprograms, Local Referencing Environments, Parameter Passing Methods, Subprograms as Parameters, Overloaded Subprograms, Closures, Co-routines

MODULE 4 (7 hours)

Support for Object Oriented Programming in different Programming languages:

Support for Object Oriented Programming – Inheritance, Dynamic Binding, Design Issues for Object Oriented Languages, Support for Object Oriented Programming in C++, Implementation of Object-oriented Constructs. Exception Handling – Basic Concepts, Design Issues.

MODULE 5 (7 hours)

Concurrency, Functional and Logic Programming:

Concurrency – Subprogram Level Concurrency, Semaphores, Monitors, Message Passing. Functional Programming Languages – Introduction to LISP and Scheme, Comparison of Functional and Imperative Languages. Logic Programming Languages – Basic Elements of Prolog, Applications of Logic Programming.

Case Studies and Modern Language Trends: Scripting vs. Compiled: Trade-offs in Python vs. C++, Dynamic Typing issues. Introduction to Rust: The Ownership Model and Borrowing (contrasting with C++ pointers).

Text Books

1. Robert W Sebesta, Concepts of Programming Languages, 10th Edition, Pearson.
2. Scott M L, Programming Language Pragmatics, 3rd Edition, Morgan Kauffman Publishers.

Reference Books

1. Kenneth C. Loudon, Programming Languages: Principles and Practice, 2nd Edition, Cengage Learning.
2. Tucker A. B. and R. E. Noonan, Programming Languages: Principles and Paradigms, 2nd Edition. –TMH.
3. Ravi Sethi, Programming Languages: Concepts and Constructs, 2nd Edition., Pearson Education.
4. David A. Watt, Programming Language Design Concepts, Wiley Dreamtech.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to Programming Paradigms		7 hours
1.1	Introduction: Reasons for studying Concepts of programming languages, Programming Domains	1
1.2	Language Evaluation Criteria	1
1.3	Influence on Language Design, Language Design Trade-offs	1
1.4	Implementation Methods	1

1.5	Names, Variables	1
1.6	Concept of Binding	1
1.7	Scope and Lifetime, Referencing Environments	1
Module 2: Data Types and Expressions in major programming languages		7 hours
2.1	Primitive Data Types, Character String Types	1
2.2	User-Defined Ordinal Types, Array Types	1
2.3	Record Types, List Types, Pointer and Reference Types	1
2.4	Implementation of pointer and reference types, Type Checking, Strong Typing, Type Equivalence	1
2.5	Expressions and Assignment Statements, Arithmetic Expressions	1
2.6	Overloaded Operators, Type Conversions	1
2.7	Relational and Boolean Expressions, Short-Circuit Evaluation, Assignment Statements, Mixed-mode Assignment	1
Module 3: Flow Control and Subprogram Concepts		8 hours
3.1	Selection Statements, Iterative Statements	1
3.2	Unconditional Branching	1
3.3	Guarded Commands	1
3.4	Subprograms: Design Issues of Subprograms	1
3.5	Local Referencing Environments	1
3.6	Parameter Passing Methods	1
3.7	Subprograms as Parameters, Overloaded Subprograms	1
3.8	Closures, Co-routines	1
Module 4: Support for OOP in different Programming languages		7 hours
4.1	Inheritance	1
4.2	Dynamic Binding	1
4.3	Design Issues for Object Oriented Languages	1
4.4	Support for Object Oriented Programming in C++	1
4.5	Implementation of Object-Oriented Constructs	1
4.6	Exception Handling – Basic Concepts	1

4.7	Exception Handling - Design Issues	1
Module 5: Concurrency, Functional and Logic Programming		7 hours
5.1	Subprogram Level Concurrency, Semaphores, Monitors, Message Passing	1
5.2	Introduction to LISP and Scheme	1
5.3	Comparison of Functional and Imperative Languages	1
5.4	Basic Elements of Prolog,	1
5.5	Applications of Logic Programming	1
5.6	Scripting vs. Compiled: Trade-offs in Python vs. C++, Dynamic Typing issues.	1
5.7	Introduction to Rust: The Ownership Model and Borrowing (contrasting with C++ pointers).	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Compare any three programming languages based on the language evaluation criteria. Prepare a list of characteristics that affect the language evaluation criteria.
2. Identify the advantages and disadvantages of imperative, functional and logic programming languages.

Course Outcome 2 (CO 2):

1. Two most important design issues that are specific to character string types are
(1) whether a string is simply a special kind of character array or a primitive type.
(2) whether strings have static or dynamic length.
Identify the implementations options for the above two cases.
2. Consider the following records of a particular language. Let the size of each char variable be 1 byte, int be 4 bytes and and Boolean be 1 bit.

```
struct Student
{
    int id;
    char name[2];
    int age;
    boolean scholarship;
}
```

Draw and comment on the possible memory layouts for the record for a 32-bit aligned machine

Course Outcome 3 (CO 3):

1. Explain three situations where a combined counting and logical looping statement is needed.
2. Describe the ways that aliases can occur with pass-by-reference parameters.
3. Identify the two fundamental design considerations for parameter-passing methods.
4. What will be the output of the given program segment if it uses the following parameter passing mechanisms:
 - a) call by reference
 - b) call by value

```
x : integer      -- global

procedure foo(y : integer)
    y := 3
    print x
    . . .

x := 2
foo(x)
print x
```

Course Outcome 4 (CO 4):

1. Describe the role of a virtual method table in implementing dynamic method binding.
2. Identify the merits and demerits of inheritance.

Course Outcome 5 (CO 5):

1. Evaluate the use of semaphores and monitors for providing competition synchronization and cooperation synchronization.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CS3P11

Course Name: PROGRAMMING PARADIGMS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Differentiate between readability and writability.
2. Define binding and binding time.
3. What are the advantages of user-defined enumeration types?
4. Define narrowing and widening conversions.
5. Why for statement in C language is more flexible than that of older languages?
6. What are the advantages and disadvantages of dynamic local variables in subprograms?
7. Illustrate the concept of dynamic method binding with an example.
8. Is it mandatory to use constructors in object-oriented languages? Justify your answer.
9. What are the applications of logic programming languages?
10. Explain the working of let and let-rec constructs in Scheme.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain different criteria used for evaluating languages. (7)
(b) Consider the following pseudocode: (7)

```
x : integer := 3
y : integer := 4

procedure add
  x := x + y

procedure second(P : procedure)
  x : integer := 5
  P()

procedure first
  y : integer := 6
  second(add)

first()
write integer(x)
```

- i. What does this program print if the language uses static scoping? Give reasons.
- ii. What does it print if the language uses dynamic scoping? Give reasons.

OR

12. (a) With respect to storage binding, explain the meanings, purposes, advantages and disadvantages of four categories of scalar variables. (7)
(b) What is meant by referencing environment of a statement? Show the referencing environment at the indicated program points (1), (2), (3) & (4) for the below given program segment. Assume that the programming language is statically scoped. (7)

```
program example;
var a, b : integer;

procedure sub1;
var x, y: integer;
begin { sub1 }
  ..... (1)
end { sub1 }

procedure sub2;
var x : integer;
  .....
  procedure sub3;
  var x: integer;
  begin { sub3 }
    ..... (2)
  end { sub3 }
```

```
begin { sub2 }  
..... (3)  
end { sub2 }
```

```
begin { example }  
..... (4)  
end { example }
```

13. (a) Explain any two issues associated with the pointer data types and also indicate how dangling pointer problem can be solved. (7)
(b) Describe the lazy and eager approaches for reclaiming garbage. (7)

OR

14. (a) What is meant by side effect and illustrate the advantages of referential transparency? (8)
(b) Explain the terms: compound assignment operator, coercion and short circuit evaluation. (6)
15. (a) Illustrate the different categories of iteration control statements. (8)
(b) Explain the techniques used for identifying the correct referencing environment for a subprogram that was sent as a parameter. (6)

OR

16. (a) Describe the implementation models of Parameter passing. (10)
(b) Differentiate coroutines from conventional subprograms. (4)
17. (a) What is meant by an exception handler? Explain how exceptions are handled in object-oriented languages. (7)
(b) Describe the design issues in object-oriented languages. (7)

OR

18. (a) Illustrate how a virtual method table can be used for implementing dynamic method binding. (7)
(b) Explain the different categories, merits and demerits of inheritance. (7)
19. (a) Compare functional and imperative programming languages. (7)
(b) Explain the role of monitors in concurrency. (7)

OR

20. (a) Explain the searching strategies used in Prolog. Why backward chaining is preferred over forward chaining in Prolog? (10)
(b) Write the output of the below code? Explain how let and lambda construct works? (4)

```
(let ((a 6)  
      (b 8)  
      (square (lambda (x) (* x x)))  
      (plus +))  
      (sqrt (plus (square a) (square b))))
```

B24CS3P12	NUMBER THEORY	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4		2024

Preamble

The purpose of this course is to create awareness among learners about the important areas of number theory used in computer science. This course covers Divisibility & Modular Arithmetic, Primes & Congruences, Euler's Function, Quadratic Residues and Arithmetic Functions, Sum of Squares and Continued fractions. Concepts in Number Theory help the learner to apply them eventually in practical applications in Computer organization & Security, Coding & Cryptography, Random number generation, Hash functions and Graphics.

Prerequisites

Mathematics

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Understand divisibility, modular arithmetic and apply algorithms such as Euclid and extended Euclid.(Cognitive Knowledge Level: Understand)
CO 2	Explain prime numbers, congruence relations and related theorems (Cognitive Knowledge Level: Understand)
CO 3	Apply number theory concepts in cryptosystems, primitive roots and classical computations (Cognitive Knowledge Level: Apply)
CO 4	Analyse quadratic residues, Legendre/Jacobi symbols and arithmetic functions including Möbius inversion. (Cognitive Knowledge Level: Apply)
CO 5	Apply number theory techniques to solve computational problems and analyze their broader implications. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	1				1					
CO2	2	1			1						
CO3	2	3	1	1	1						1
CO4	2	2		1							
CO5	2	2		1	1	2	1				1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 HOURS)

Divisibility and Modular Arithmetic: Finite Fields – Groups, Rings and Fields. Divisibility - Divisibility and Division Algorithms, Well ordering Principle, Bezout's Identity. Modular Arithmetic- Properties, Euclid's algorithm for the greatest common divisor, Extended Euclid's Algorithm, Least Common multiple, Solving Linear Diophantine Equations, Modular Division.

MODULE 2 (7 HOURS)

Primes and Congruences: Prime Numbers-Prime Numbers and prime-power factorization, Fermat and Mersenne primes., Primality testing and factorization. Computer Science and Engineering (Data Science) Congruences-Linear congruences, Simultaneous linear congruences, Chinese Remainder Theorem, Fermat's little theorem, Wilson's theorem.

MODULE 3 (8 HOURS)

Congruences with a Prime-Power Modulus & Euler's Function: Congruences with a Prime-Power Modulus-Arithmetic modulo p , Pseudoprimes and Carmichael numbers, Solving congruences modulo prime powers. Euler's Function-Euler's Totient function, Applications of Euler's Totient function, Traditional Cryptosystem, Limitations. The Group of units- The group U_n , Primitive roots, Existence of primitive roots, Applications of primitive roots.

MODULE 4 (7 HOURS)

Quadratic Residues & Arithmetic Functions : Quadratic Residues- Quadratic Congruences, The group of Quadratic residues, Legendre symbol, Jacobi Symbol, Quadratic reciprocity. Arithmetic Functions- Definition and examples, Perfect numbers, Mobius function and its properties, Mobius inversion formula, The Dirichlet Products.

MODULE 5 (7 HOURS)

Sum of Squares and Continued Fractions: Sum of Squares- Sum of two squares, The Gaussian Integers, Sum of three squares, Sum of four squares. Continued Fractions -Finite continued fractions, Infinite continued fractions, Pell's Equation, Solution of Pell's equation by continued fractions

Text Books

1. G.A. Jones & J.M. Jones, Elementary Number Theory, Springer UTM, 2007.

2. Joseph Silverman, A Friendly introduction to Number Theory, Pearson Ed. 2009.

Reference Books

1. William Stallings, Cryptography and Network Security Principles and Practice, Pearson Ed.
2. Tom M.Apostol, 'Introduction to Analytic Number Theory', Narosa Publishing House Pvt. Ltd, New Delhi, (1996).
3. Neal Koblitz, A course in Number Theory and Cryptography, 2nd Edition, Springer ,2004.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
MODULE 1: Divisibility and Modular Arithmetic		7 Hours
1.1	Groups, Rings and Fields	1
1.2	Divisibility, Division Algorithm	1
1.3	Well Ordering Principle, Bezout's Identity	1
1.4	Modular Arithmetic and Properties	1
1.5	Euclid's Algorithm and Extended Euclid's Algorithm	1
1.6	LCM and Linear Diophantine Equations	1
1.7	Modular Division	1
MODULE 2: Primes and Congruences		7 Hours)
2.1	Prime Numbers and Prime-power Factorization	1
2.2	Fermat and Mersenne Primes	1
2.3	Primality Testing and Factorization	2
2.4	Linear and Simultaneous Congruences	1
2.5	Chinese Remainder Theorem	1
2.6	Fermat's Little Theorem and Wilson's Theorem	1

MODULE 3: Prime-Power Modulus & Euler's Function		8 Hours
3.1	Arithmetic Modulo p, Pseudoprimes, Carmichael Numbers	2
3.2	Solving Congruences Modulo Prime Powers	2
3.3	Euler's Totient Function and Applications	2
3.4	Traditional Cryptosystems and Limitations	1
3.5	Group of Units U_n and Primitive Roots	1
MODULE 4: Quadratic Residues and Arithmetic Functions		7 Hours
4.1	Quadratic Congruences and Group of Quadratic Residues	1
4.2	Legendre and Jacobi Symbols	2
4.3	Quadratic Reciprocity	2
4.4	Arithmetic Functions: Möbius, Perfect Numbers	1
4.5	Möbius Inversion	1
MODULE 5: Sum of Squares and Continued Fractions		7 Hours
5.1	Sum of Two Squares and Gaussian Integers	2
5.2	Sum of Three and Four Squares	1
5.3	Finite Continued Fractions	1
5.4	Infinite Continued Fractions	1
5.5	Pell's Equation and Solutions via Continued Fractions	2

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Using the Extended Euclidean Algorithm, compute the modular inverse of $47 \pmod{64}$ and verify your result.
2. Solve the linear Diophantine equation $35x + 10y = 5$.

Course Outcome 2 (CO 2):

1. Discuss Wilson's theorem and prove that 6 is not prime using it.
2. Apply the Chinese Remainder Theorem to solve the system of congruences:

$$x \equiv 4 \pmod{7}, \quad x \equiv 5 \pmod{9}, \quad x \equiv 2 \pmod{11}.$$

Course Outcome 3 (CO 3):

1. Apply the Carmichael number criteria to demonstrate that 561 is a Carmichael number.
2. Explain the role of primitive roots in classical cryptosystems.

Course Outcome 4 (CO 4):

1. Define Quadratic Residues. Find quadratic residues and non-residues of 11
2. Solve the polynomial congruence

$$x^3 + 4x \equiv 4 \pmod{343},$$

where $343 = 7^3$.

Hint: First find solutions modulo 7 (and modulo 7^2), then lift them to modulo 7^3 (e.g. using Hensel's lemma).

Course Outcome 5 (CO 5):

1. Use the extended Euclid's algorithm to solve Diophantine equations efficiently. Given three numbers $a > 0$, $b > 0$, and c , the algorithm should return some integers x and y such that
$$ax + by = c.$$
2. Solve Pell's equation $x^2 - 7y^2 = 1$ using continued fractions.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CS3P12

Course Name: NUMBER THEORY

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. State the principle of well ordering.
2. Apply Euclidean Algorithm to compute the GCD of (270,192).
3. Define Fermat prime and Mersenne prime with example.
4. Use Fermat's little theorem to find: $2925 \bmod 11$.
5. Outline the concept of Carmichael number. Give an example.
6. List out the applications of primitive roots.
7. Define Mobius inversion formula.
8. Define quadratic residues and non-residues modulo a prime p . Find all quadratic residues modulo 11.
9. Explain the theorem for sum of two squares and sum of three squares.
10. State Pell's equation.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Summarize the properties of Modulo arithmetic and Modulo operator. (7)
(b) Apply Euclidean algorithm to find out the $\gcd(1492, 1066)$ and express it in terms of Bezout's identity. (7)

OR

12. (a) Find all the solutions of $5x + 3y = 4$, using Linear Diophantine equations. (7)
(b) State the relationship between GCD, LCM and product of two numbers a and b . Use it to find the LCM of $[1050, 2574]$. (7)
13. (a) Use Chinese Remainder Theorem to solve the simultaneous congruence:
$$x \equiv 6 \pmod{11}, \quad x \equiv 13 \pmod{16}, \quad x \equiv 9 \pmod{21}, \quad x \equiv 19 \pmod{25}$$

(9)

- (b) State Wilson's Theorem. Compute the value of $(97!) \pmod{101}$ using Wilson's theorem. (5)

OR

14. (a) Explain the concept of Fermat's factorization theorem and use it to factorize the number 5959. (7)
(b) State Fermat's Little theorem. Show how computations can be simplified using Fermat's Little theorem by computing the value of $2^{35} \pmod{7}$. (7)
15. (a) With the aid of a figure, explain the concept of Symmetric key Encryption. Also mention its pros and cons. (10)
(b) What are the challenges faced in public key crypto systems? (4)

OR

16. (a) When can we say that an integer n is a pseudoprime? Check whether the number 341 is pseudo prime number. (6)
(b) Explain Euler Totient function with example. What is the value of $\phi(n)$ when n is a composite number and use it to compute (i) $\phi(240)$ (ii) $\phi(49)$. (8)
17. (a) State the generalised law of Quadratic reciprocity and hence evaluate the Jacobi symbol $\left(\frac{221}{399}\right)$. (8)
(b) Prove that 45 is not a quadratic residue modulo 47. (6)

OR

18. (a) State and prove the properties of Legendre Symbol (6)
(b) Find the following values of the Legendre symbols:

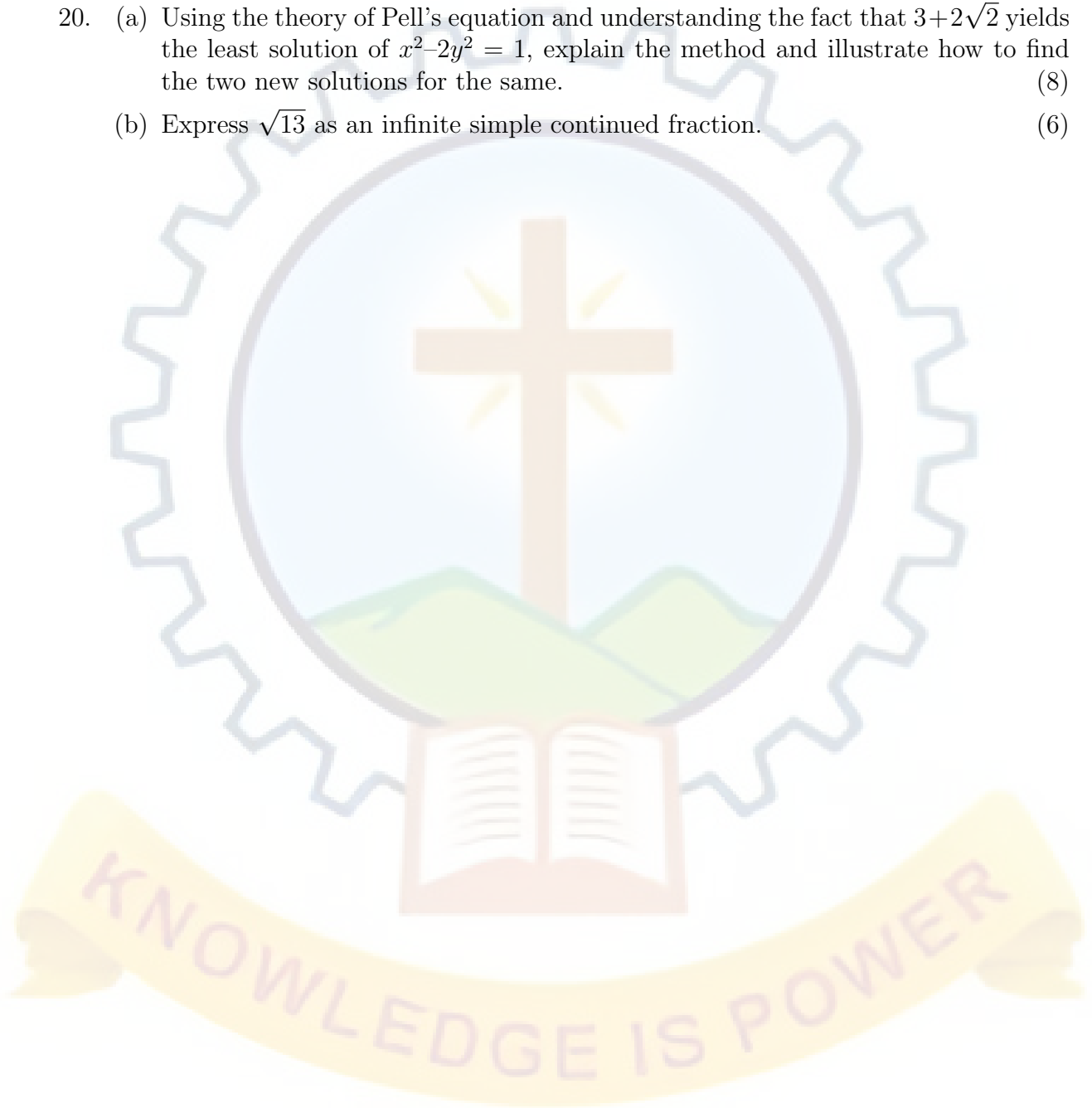
$$\left(\frac{113}{127}\right), \left(\frac{113}{131}\right), \left(\frac{113}{137}\right), \left(\frac{210}{229}\right)$$

(8)

19. (a) Express 28 as a sum of four squares. (6)
(b) Apply the theorem of Gaussian Integer unique factorization for the numbers 45 and 65 to count in how many different ways they can be written as a sum of two squares. (8)

OR

20. (a) Using the theory of Pell's equation and understanding the fact that $3+2\sqrt{2}$ yields the least solution of $x^2-2y^2 = 1$, explain the method and illustrate how to find the two new solutions for the same. (8)
(b) Express $\sqrt{13}$ as an infinite simple continued fraction. (6)



B24CS3P13	INTERNET OF THINGS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4		

Preamble

This course provides a foundational understanding of the Internet of Things (IoT), emphasizing its principles, architecture, and role in building interconnected intelligent systems. It covers IoT devices, communication technologies, application protocols, data analytics, and practical system development using Raspberry Pi, Python, and cloud platforms. The course equips students with the technical competencies required to design, implement, and utilize IoT solutions, enabling them to meet future demands in smart environments, automation, and emerging digital technologies.

Prerequisites

Basic knowledge in Computer Networks

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the fundamentals and architecture of IoT systems.(Cognitive Knowledge Level: Understand)
CO 2	Describe IoT hardware components, sensors, actuators, and communication technologies. (Cognitive Knowledge Level: Understand)
CO 3	Analyze key IoT communication protocols such as MQTT and CoAP.(Cognitive Knowledge Level: Apply)
CO 4	Apply data analytics techniques for IoT applications using edge and cloud-based tools. (Cognitive Knowledge Level: Apply)
CO 5	Design simple IoT applications using Raspberry Pi, REST APIs, and cloud platforms considering real-world applications and responsible use. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2		1					1		
CO 2	2	3	1	2	1				1		
CO 3	3	3	2	2	2				1		2
CO 4	3	3	3	3	3	1			2		3
CO 5	3	3	3	3	3	3	2	2	3	2	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	60	60	60
Apply	10	10	10
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Fundamentals of IoT

What is IoT, Genesis of IoT and IoT Impact, Digitization and Convergence of IT and IoT, IoT Challenges, IoT Network Architecture and Design, The Core IoT Functional Stack, IoT Data Management and Compute Stack

MODULE 2 (7 hours)

Smart Objects and Connectivity

Smart Objects – the “Things” in IoT, Sensors, Actuators, and Smart Objects, Sensor Networks and Connectivity Models, Communications Criteria, IoT Access Technologies (IEEE 802.15.4, 802.11ah, LoRa, NB-IoT)

MODULE 3 (7 hours)

IP and Application Protocols for IoT

IP as the IoT Network Layer and Need for Optimization, Optimizing IP for IoT – Profiles and Compliances, Application Protocols for IoT – MQTT and CoAP, Transport Layer and IoT Application Transport Methods

MODULE 4 (6 hours)

Data and Analytics for IoT

Data and Analytics for IoT-Nature of IoT data, Types of IoT data, IoT data life cycle, Introduction to Machine Learning for IoT-Role of machine learning, Types of ML used in IoT, Big Data Analytics Tools and Technologies, Edge and Streaming Analytics, Network Analytics and Visualization and Interpretation of IoT data.

MODULE 5 (10 hours)

Developing IoT Systems

IoT Logical Design using Python, IoT Physical Devices and Endpoints – Raspberry Pi Interfaces, Programming Raspberry Pi with Python, Cloud Storage Models and IoT Cloud Offerings, Designing RESTful Web APIs, Cloud Web Services for IoT (AWS IoT / ThingSpeak)

***Students must complete one micro-project related to any IoT concept covered in the course modules, demonstrating practical understanding and application of theoretical topics.**

Text Books

1. David Hanes et al., IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things, Cisco Press / Pearson Education.
2. Arshdeep Bahga & Vijay Madisetti, Internet of Things: A Hands-On Approach, University Press, 2015.

Reference Books

1. Rajkamal, Internet of Things: Architecture and Design Principles, McGraw Hill.
2. Ovidiu Vermesan & Peter Friess, Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems, River Publishers.
3. Simon Monk, Programming Raspberry Pi: Getting Started with Python, McGraw Hill.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to IoT and Architecture		6 hours
1.1	What is IoT – Genesis and Evolution of IoT	1
1.2	IoT and Digitization, IoT Impact and Opportunities	1
1.3	Convergence of IT and OT, IoT Challenges	1
1.4	IoT Network Architecture and Design	1
1.5	Comparing IoT Architectures, Simplified IoT Architecture	1
1.6	The Core IoT Functional Stack, Data Management and Compute Stack	1
Module 2: Smart Objects and Connectivity		7 hours
2.1	Smart Objects – “The Things” in IoT	1
2.2	Sensors, Actuators, and Smart Object Characteristics	1
2.3	Sensor Networks and Connectivity Models	1
2.4	Connecting Smart Objects and Communication Criteria	1
2.5	IoT Access Technologies – IEEE 802.15.4	1

2.6	IoT Access Technologies – 802.11ah and LoRaWAN	1
2.7	IoT Access Technologies – NB-IoT and LTE	1
Module 3:IP and Application Protocols for IoT		7 hours
3.1	IP as the IoT Network Layer and Business Case for IP	1
3.2	Need for Optimization – Constrained Devices and Networks	1
3.3	Optimizing IP for IoT – Profiles and Compliance	1
3.4	Application Protocols for IoT – CoAP (Constrained Application Protocol)	2
3.5	Application Protocols for IoT – MQTT (Message Queuing Telemetry Transport)	1
3.6	Transport Layer – IoT Application Transport Methods	1
Module 4: Data and Analytics for IoT		6 hours
4.1	Data and Analytics for IoT – Overview	1
4.2	Introduction to Machine Learning for IoT	1
4.3	Big Data Analytics Tools and Technologies	1
4.4	Edge Streaming Analytics and Network Analytics	1
4.5	Visualization and Interpretation of IoT Data	1
4.6	IoT Security Overview – Common Challenges and Risk Analysis Concepts	1
Module 5: Developing IoT Systems		10 hours
5.1	IoT Logical Design using Python	1
5.2	IoT Physical Devices and Endpoints – Raspberry Pi Interfaces	1
5.3	Programming Raspberry Pi using Python	1
5.4	Interfacing Sensors and Actuators with Raspberry Pi	1
5.5	Cloud Storage Models and IoT Cloud Offerings	1
5.6	Designing RESTful Web APIs for IoT Applications	1
5.7	Cloud Web Services for IoT (AWS IoT / ThingSpeak)	1
5.8	Case Study – Smart Home / Smart Agriculture Application	1
5.9	Security and Privacy in IoT System Implementation	1
5.10	Project Discussion / Demonstration	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the convergence of IT and OT in the context of IoT and describe how digitization impacts this convergence.
2. Illustrate the simplified IoT architecture and discuss the role of the Core IoT Functional Stack.
3. Compare different IoT reference architectures and identify key challenges in their implementation.

Course Outcome 2 (CO 2):

1. Describe the characteristics and functions of smart objects, sensors, and actuators used in IoT.
2. Differentiate between IEEE 802.15.4, LoRaWAN, and NB-IoT technologies in terms of range, bandwidth, and power usage.
3. Explain the process of connecting smart objects to the IoT network using appropriate communication criteria.

Course Outcome 3 (CO 3):

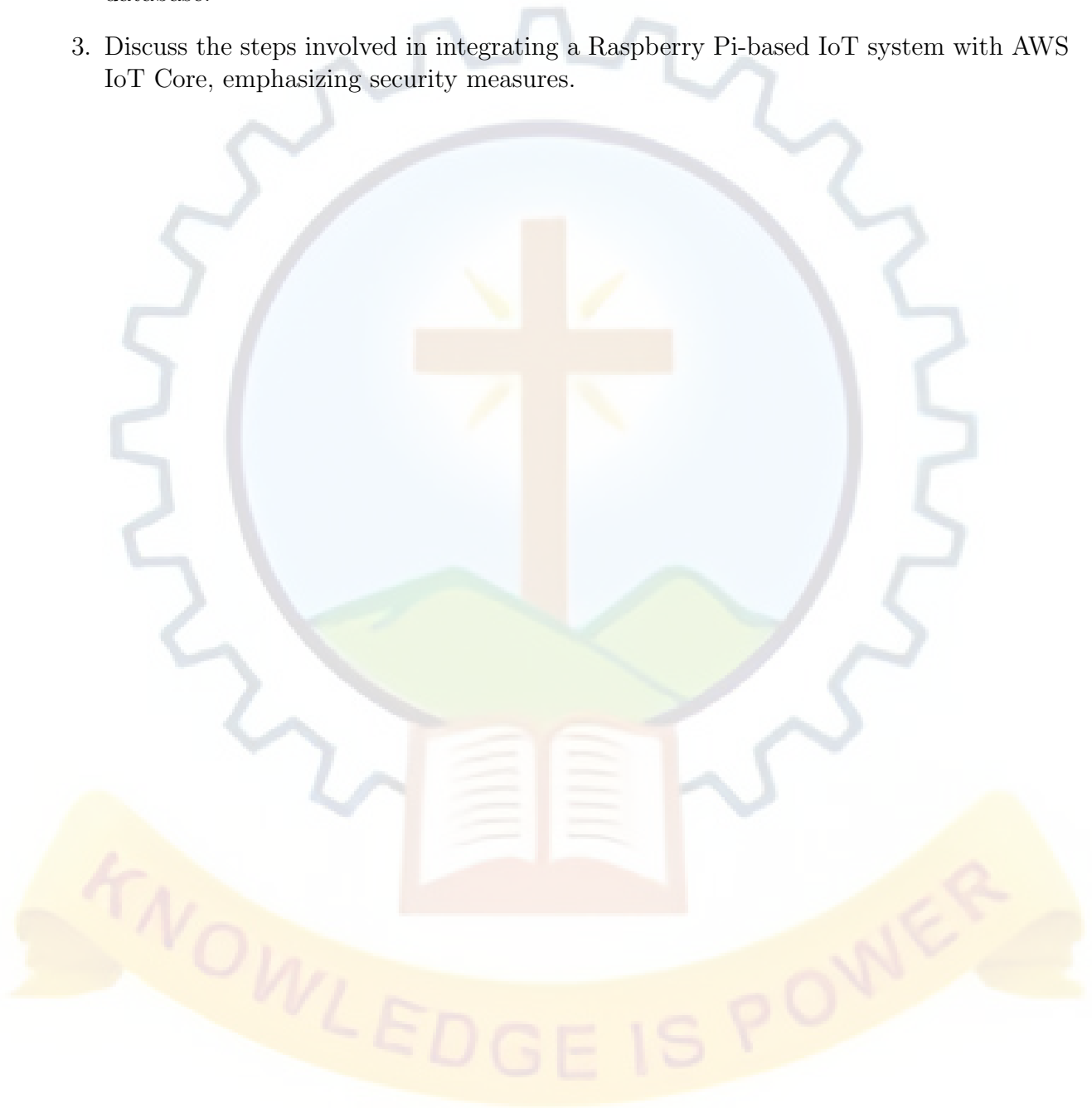
1. Analyze the advantages and limitations of using MQTT versus CoAP in constrained IoT networks.
2. Explain how IP is optimized for IoT and discuss the need for lightweight transport mechanisms.
3. Demonstrate (using a case) how the transport layer supports message delivery reliability in IoT applications.

Course Outcome 4 (CO 4):

1. Apply big data analytics concepts to an IoT scenario such as smart energy management.
2. Explain the role of edge analytics in reducing latency for IoT data processing.
3. Interpret IoT sensor data using suitable visualization techniques and justify the selected tool.

Course Outcome 5 (CO 5):

1. Design a simple IoT application using Raspberry Pi to monitor temperature and upload readings to a cloud platform (e.g., ThingSpeak).
2. Develop a RESTful Web API that interfaces a physical IoT device with a cloud database.
3. Discuss the steps involved in integrating a Raspberry Pi-based IoT system with AWS IoT Core, emphasizing security measures.



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CS3P13

Course Name: INTERNET OF THINGS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define IoT. Explain the impact of IoT on digitization and convergence of IT and OT.
2. Describe the major challenges faced in large-scale IoT deployment.
3. Differentiate between sensors and actuators with suitable examples.
4. Explain the advantages of using LoRaWAN in IoT communication.
5. What are the major communication criteria to be considered while connecting smart objects?
6. Compare MQTT and CoAP in terms of architecture and communication style.
7. Describe the need for optimizing IP for IoT networks.
8. What is edge analytics? How does it differ from cloud-based analytics?
9. Explain the importance of data visualization in IoT applications.
10. List the major components of a Raspberry Pi-based IoT system and their functions.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the Core IoT Functional Stack with neat diagram. (7)
(b) Discuss IoT Network Architecture and compare different IoT architectural models. (7)

OR

12. (a) Explain the Genesis and Evolution of IoT . (6)
(b) Describe the IoT Data Management and Compute Stack and its importance. (8)
13. (a) Describe the role of smart objects in IoT with examples. (7)
(b) Explain various IoT access technologies and their comparative advantages. (7)

OR

14. (a) Explain the communication criteria for connecting smart objects. (6)
(b) Compare IEEE 802.15.4, LoRaWAN, and NB-IoT technologies. (8)
15. (a) Discuss how IP functions as the IoT network layer. (6)
(b) Compare MQTT and CoAP protocols with respect to reliability, overhead, and use cases. (8)

OR

16. (a) Explain the need for optimization of IP for constrained devices and networks. (7)
(b) Describe the working of the transport layer in IoT communication. (7)
17. (a) Explain the role of data analytics in IoT systems. (7)
(b) Discuss the concept of edge streaming analytics with an example. (7)

OR

18. (a) Explain how machine learning techniques are applied in IoT data analysis. (8)
(b) Describe the major big data analytics tools and technologies used for IoT. (6)
19. (a) Demonstrate how Raspberry Pi can be programmed to interface a temperature sensor using Python. (7)
(b) Explain the design and development of a RESTful web API for IoT data exchange. (7)

OR

20. (a) Discuss the architecture and functioning of cloud web services for IoT (e.g., AWS IoT or ThingSpeak). (8)
(b) Explain the essential security and privacy considerations in IoT application development. (6)

B24CS3P14	DISTRIBUTED COMPUTING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

The purpose of this course is to understand the system models, algorithms and protocols that allow computers to communicate and coordinate their actions to solve a problem. This course helps the learner to understand the distributed computation model and various concepts like global state, termination detection, mutual exclusion, deadlock detection, shared memory, failure recovery, consensus, file system. It helps the learners to develop solutions to problems in distributed computing environment.

Prerequisites

Basic knowledge in data structures and operating systems.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Summarize various aspects of distributed computation model and logical time. (Cognitive Knowledge Level: Understand)
CO 2	Illustrate election algorithm, global snapshot algorithm and termination detection algorithm. (Cognitive Knowledge Level: Apply)
CO 3	Compare token based, non-token based and quorum based mutual exclusion algorithms. (Cognitive Knowledge Level: Apply)
CO 4	Recognize the significance of deadlock detection and shared memory in distributed systems. (Cognitive Knowledge Level: Apply)
CO 5	Explain the concepts of failure recovery and consensus. (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1	0	0							
CO 2	2	2	1	2							
CO 3	2	2	1	2							
CO 4	2	2	1	2							
CO 5	2	2	1	2		2			1		

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

(Distributed systems basics and Computation model)

Distributed System – Definition, Relation to computer system components, Motivation, Primitives for distributed communication, Design issues, Challenges and applications. A model of distributed computations – Distributed program, Model of distributed executions, Models of communication networks, Global state of a distributed system, Cuts of a distributed computation, Past and future cones of an event, Models of process communications.

MODULE 2 (7 hours)

Election algorithm, Global state and Termination detection

Logical time – A framework for a system of logical clocks, Scalar time, Vector time. Leader election algorithm – Bully algorithm, Ring algorithm. Global state and snapshot recording algorithms – System model and definitions, Snapshot algorithm for FIFO channels – Chandy Lamport algorithm. Termination detection – System model of a distributed computation, Termination detection using distributed snapshots, Termination detection by weight throwing, Spanning-tree-based algorithm.

MODULE 3 (7 hours)

Mutual exclusion and Deadlock detection

Distributed mutual exclusion algorithms – System model, Requirements of mutual exclusion algorithm. Lamport's algorithm, Ricart-Agrawala algorithm, Quorum-based mutual exclusion algorithms – Maekawa's algorithm. Token-based algorithm – Suzuki-Kasami's broadcast algorithm. Deadlock detection in distributed systems – System model, Deadlock handling strategies, Issues in deadlock detection, Models of deadlocks.

MODULE 4 (8 hours)

Distributed shared memory and Failure recovery

Distributed shared memory – Abstraction and advantages. Shared memory mutual exclusion – Lamport's bakery algorithm. Check pointing and rollback recovery – System model, consistent and inconsistent states, different types of messages, Issues in failure recovery, checkpoint based recovery, log based roll back recovery.

MODULE 5 (7 hours)

Consensus and Distributed file system

Consensus Consensus and agreement algorithms – Assumptions, The Byzantine agreement and other problems, Agreement in (message-passing) synchronous systems with failures –

Consensus algorithm for crash failures. Distributed file system – File service architecture, Case studies: Sun Network File System, Andrew File System, Google File System.

Textbook

1. Ajay D. Kshemkalyani and Mukesh Singhal, Distributed Computing: Principles, Algorithms, and Systems, Cambridge University Press, 2011.

Reference Books

1. George Coulouris, Jean Dollimore, Tim Kindberg and Gordon Blair. Distributed Systems: Concepts and Design, Addison Wesley, Fifth edition.
2. Kai Hwang, Geoffrey C Fox, Jack J Dongarra, Distributed and Cloud Computing – From Parallel Processing to the Internet of Things, Morgan Kaufmann Publishers, 2012.
3. Sukumar Ghosh, Distributed Systems: An Algorithmic Approach, CRC Press, Second edition, 2015.
4. Maarten Van Steen, Andrew S. Tanenbaum, Distributed Systems, Prentice Hall of India, Third edition, 2017.
5. Randy Chow and Theodore Johnson, Distributed Operating Systems and Algorithm Analysis, Pearson Education India, First edition, 2009.
6. Valmir C. Barbosa, An Introduction to Distributed Algorithms, MIT Press, 2003.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Distributed systems basics and Computation model		7 hours
1.1	Distributed System – Definition, Relation to computer system components	1
1.2	Primitives for distributed communication.	1
1.3	Design issues	1

1.4	challenges, Applications.	1
1.5	A model of distributed computations – Distributed program, Model of distributed executions	1
1.6	Models of communication networks, Global state of a distributed system,	1
1.7	Cuts of a distributed computation, Past and future cones of an event, Models of process communications.	1
Module 2: Election algorithm, Global state and Termination detection		7 hours
2.1	Logical time – A framework for a system of logical clocks, Scalar time	1
2.2	Vector time.	1
2.3	Leader election algorithm – Bully Algorithm, Ring Algorithm	1
2.4	Global state and snapshot recording algorithms – System model and definitions	1
2.5	Snapshot algorithm for FIFO channels – Chandy Lamport algorithm.	1
2.6	Termination detection – System model of a distributed computation, Termination detection using distributed snapshots	1
2.7	Termination detection by weight throwing , Spanning tree-based algorithm	1
Module 3: Mutual exclusion and Deadlock detection		7 hours
3.1	Distributed mutual exclusion algorithms – System model , Lamport's algorithm	1
3.2	Ricart–Agrawala algorithm	1
3.3	Quorum-based mutual exclusion algorithms – Maekawa's algorithm	1
3.4	Token-based algorithm – Suzuki–Kasami's broadcast algorithm.	1
3.5	Deadlock detection in distributed systems – System model, Deadlock handling strategies	1
3.6	Issues in deadlock detection	1
3.7	Models of deadlocks	1
Module 4: Distributed shared memory and Failure recovery		8 hours

4.1	Distributed shared memory – Abstraction and advantages.	1
4.2	Shared memory mutual exclusion – Lamport's bakery algorithm.	1
4.3	Checkpointing and rollback recovery – System model, consistent and inconsistent states	1
4.4	Different types of messages	1
4.5	Issues in failure recovery	1
4.6	checkpoint based recovery	1
4.7	checkpoint based recovery	1
4.8	log based roll back recovery.	1
Module 5: Consensus and Distributed file system		7 hours
5.1	Consensus and agreement algorithms – Assumptions	1
5.2	The Byzantine agreement and other problems	1
5.3	Agreement in (message-passing) synchronous systems with failures – Consensus algorithm for crash failures	1
5.4	Agreement in (message-passing) synchronous systems with failures – Consensus algorithm for crash failures	1
5.5	Distributed File System – File Service Architecture	1
5.6	Case Studies: Sun Network File System	1
5.7	Andrew File System, Google File System.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Define logical clock and explain the implementation of the logical clock.
2. Explain different forms of load balancing.

Course Outcome 2 (CO 2):

1. Apply ring-based leader election algorithm with 10 processes in the worst-performing case. Count the number of messages needed.

2. Apply spanning tree-based termination detection algorithm in the following scenario. The nodes are processes 0 to 6. Leaf nodes 3, 4, 5, and 6 are each given tokens T3, T4, T5 and T6 respectively. Leaf nodes 3, 4, 5 and 6 terminate in the order, but before terminating node 5, it sends a message to node 1.

Course Outcome 3 (CO 3):

1. What are the requirements of mutual exclusion algorithms?
2. Illustrate Suzuki–Kasami’s broadcast algorithm.

Course Outcome 4 (CO 4):

1. Compare different models of deadlocks.
2. Illustrate the detailed abstraction of distributed shared memory and interaction with application processes.

Course Outcome 5 (CO 5):

1. Explain how consensus problem differs from the Byzantine agreement problem.
2. Classify different log-based rollback recovery techniques.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CS3P14

Course Name: DISTRIBUTED COMPUTING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Identify any three distributed applications and for each application, determine which all motivating factors are important for building an application over a distributed system.
2. Assume that the surface of the past cone form a consistent cut. Does it mean that all events on the surface of the past cone are always concurrent? Demonstrate with the help of an example.
3. Specify the issues in recording a global state.
4. Explain the rules used to update clocks in scalar time representation.
5. Describe how quorum-based mutual exclusion algorithms differ from the other categories of mutual exclusion algorithms.
6. Explain with example, how wait-for-graphs can be used in deadlock detection.
7. List any three advantages of using distributed shared memory.
8. Define the no-orphans consistency condition.
9. Define Byzantine agreement problem.
10. Differentiate between whole file serving and whole file caching in Andrew file system.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the three different models of the service provided by communication networks. (9)
(b) Explain how the causal dependency between events in distributed execution is defined using Lamport's happened-before relationship. (5)

OR

12. (a) Address the various strategies that can be adopted to satisfy the requirements of a reliable and fault tolerant distributed system. (5)
(b) Which are the different versions of send and receive primitives for distributed communication? Explain. (9)
13. (a) Illustrate bully algorithm for electing a new leader. Does the algorithm meet liveness and safety conditions? (8)
(b) Clearly mentioning assumptions, explain the rules of termination detection using distributed snapshots. (6)

OR

14. (a) In Chandy-Lamport algorithm for recording global snapshots, explain how the recorded local snapshots can be put together to create the global snapshot. Can multiple processes initiate the algorithm concurrently? (8)
(b) Illustrate the working of spanning-tree-based termination detection algorithm. (6)
15. (a) Explain and illustrate Lamport's mutual exclusion algorithm. (6)
(b) Discuss the three types of messages required for deadlock handling in Maekawa's algorithm. Explain how Maekawa's algorithm handles deadlocks. (8)

OR

16. (a) Explain and illustrate Ricart-Agrawala algorithm for achieving mutual exclusion. (8)
(b) Explain any three different models of deadlock. (6)
17. (a) What are the issues in failure recovery? Illustrate with suitable examples. (8)
(b) Show that Lamport's Bakery algorithm for shared memory mutual exclusion satisfies the three requirements of the critical section problem. (6)

OR

18. (a) Differentiate Andrew File System and NFS. (8)
(b) Explain Sun NFS architecture with diagram. (6)
19. (a) Differentiate consistent and inconsistent states with examples. (7)
(b) What is checkpoint-based rollback-recovery? Explain the three classifications of checkpoint-based rollback-recovery. (7)

OR

20. (a) Explain consensus algorithm for crash failures under synchronous systems. (4)
(b) Summarize distributed file system requirements. (10)

B24CS3L07	SYSTEM SOFTWARE AND COMPILER DESIGN LAB	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	3	2	2	2024

Preamble

The course aims to provide students with hands-on experience in the design and implementation of system software and compiler components. Throughout the course students will familiar with the fundamental concepts of operating systems, assemblers, loaders, macro processors, and various compiler construction tools. The system software and microprocessor lab enables students to simulate core operating system algorithms, implement basic system software components, and work with tools such as LEX and YACC to automate different phases of a compiler. Through practical exercises, students will enhance their ability to design, implement, and test essential system and compiler modules, thereby strengthening their understanding of low-level software development and system-level programming.

Prerequisites

Fundamental knowledge in Operating Systems, Problem Solving and Programming Techniques, Data Structures, Formal Languages and Automata Theory, and Compiler Design

Course Outcomes

After the completion of the course the student will be able to

CO 1	Implement fundamental system software components such as assemblers, loaders, macro processors, and essential operating system algorithms. (Cognitive Knowledge Level: Apply)
CO 2	Develop lexical analyzers and syntax analyzers using compiler construction tools such as LEX and YACC. (Cognitive Knowledge Level: Apply)
CO 3	Design and simulate automata-based models (NFA, DFA) and apply parsing techniques such as top-down and bottom-up parsing. (Cognitive Knowledge level: Apply)
CO 4	Analyze the internal working principles of system software and compiler phases, including symbol tables, intermediate code generation, optimization, and interpretation. (Cognitive Knowledge level: Analyze)
CO 5	Design and construct complete compiler components and system-level modules to meet specific software requirements in real-time application development. (Cognitive Knowledge level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	2	3	2				2	1	
CO 2	2	2	2	3	2				3	1	3
CO 3	3	3	2	3	2	1			1		1
CO 4	3	3	3	3	2	3	3	1	3		3
CO 5	2	3	3	3	3	3	3	3	3	3	3

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Evaluation	20 marks
- Viva (Self Study)	10 marks
- Output	10 marks
Lab Record	5 marks
Viva-voce/Test	15 marks

SYLLABUS

LIST OF EXPERIMENTS

1	Implement pass one of two pass assembler.
2	Implement pass two of two pass assembler.
3	Implement a single pass assembler.
4	Implement a single pass macro processor.
5	Implement an absolute loader.

6	Design and implement a lexical analyzer using C language to recognize all valid tokens in the input program. The lexical analyzer should ignore redundant spaces, tabs and newlines. It should also ignore comments.
7	Write a program to find First and Follow of any given grammar.
8	Design and implement a recursive descent parser for a given grammar.
9	Construct a Shift Reduce Parser for a given language.
10	Implement Intermediate code generation for simple expressions.
11	Implement Code generation for simple expressions.
12	Implement a Lexical Analyzer for a given program using Lex Tool.
13	Write a lex program to display the number of lines, words and characters in an input text.
14	Write a LEX Program to convert the substring abc to ABC from the given input string.
15	Write a lex program to find out total number of vowels and consonants from the given input sting.
16	Implementation of Calculator using LEX and YACC.

Reference Books

1. Leland L. Beck, System Software: An Introduction to Systems Programming, 3/E, Pearson Education Asia.
2. Peter Abel, IBM PC Assembly Language and Programming, Third Edition, Prentice Hall of India.
3. Aho A.V., Ravi Sethi and D. Ullman. Compilers – Principles Techniques and Tools, Addison Wesley, 2006.

B24CS3L08	WEB PROGRAMMING LAB	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	3	2	2	2024

Preamble

This course equips learners with the skills to design and develop complete web applications using modern front-end and back-end technologies. It covers structured webpage creation using HTML, styling through CSS (including responsive design), interactive client-side programming with JavaScript, and server-side development with Node.js, PHP, MySQL, React and MongoDB. Learners will implement real-time functionalities such as form validation, CRUD operations, and dynamic content generation, culminating in full-fledged group projects like management and enrollment systems. The course provides a comprehensive foundation for building robust, scalable, and user-centric web solutions for real-world applications.

Prerequisites

Basic knowledge of programming fundamentals and familiarity with computers and internet concepts.

Course Outcomes

After the completion of the course the student will be able to

CO 1	Design and develop structured, standards-compliant web pages using HTML5 elements, semantic tags, tables, lists, forms, media components, XML, and DTD validation. (Cognitive Knowledge Level: Apply)
CO 2	Apply Inline, Internal, and External CSS to style webpages and build responsive layouts using modern CSS properties, selectors, and positioning techniques. (Cognitive Knowledge Level: Apply)
CO 3	Implement client-side interactivity using JavaScript by manipulating arrays, objects, functions, events, DOM/BOM, decision structures, loops, and form validation. (Cognitive Knowledge Level: Apply)
CO 4	Develop dynamic server-side applications using PHP and MySQL by handling forms, executing database queries, and performing CRUD operations. (Cognitive Knowledge Level: Apply)
CO 5	Build full-stack applications using Node.js and MongoDB by creating HTTP/Express servers, routing, handling asynchronous operations, and performing CRUD actions on NoSQL collections. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1		1	1	1			2		2
CO 2	2	2	2	1	2	2			2		3
CO 3	2	3	2	3	3	3	2	1	3	1	3
CO 4	2	3	3	3	3	3	3	3	3	3	3
CO 5	2	3	3	3	3	3	3	3	3	3	3

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Evaluation	20 marks
- Viva (Self Study)	10 marks
- Output	10 marks
Lab Record	5 marks
Viva-voce/Test	15 marks

SYLLABUS

LIST OF EXPERIMENTS

1	Create a web page with all possible elements of HTML5. Design a complete HTML5 web page titled “Personal Portfolio” that includes the following: • Semantic structure using <code><header></code>, <code><nav></code>, <code><main></code>, <code><section></code>, <code><article></code>, <code><aside></code>, and <code><footer></code> • Headings (<code><h1></code> to <code><h6></code>), paragraphs, line breaks, horizontal rules, special characters • Tables with headings and multiple rows and Images with <code>alt</code> attributes and hyperlinks (internal and external) • A form with the following HTML5 input types: <code>text</code>, <code>email</code>, <code>tel</code>, <code>number</code>, <code>date</code>, <code>range</code>, <code>color</code>, <code>datalist</code>, and a submit button

	• Media: embedded audio and video with controls • Use of <code><meta></code> tags, page title, and viewport setting in the <code><head></code> • Proper indentation and comments in code
2	Cascading Style Sheets: Create a web page demonstrating the use of all three types of Cascading Style Sheets (CSS): Inline CSS, Internal CSS, and External CSS. The page should contain at least the following HTML elements: heading, paragraph, list, image, and table. Apply different styles using each type of CSS and clearly differentiate their usage in the code.
3	Develop a RESPONSIVE PAGE with HTML and CSS.
4	Programs to demonstrate JavaScript array, object and function 1. Write a JavaScript program to find the sum of array elements using function. 2. Write a JavaScript program to sort a list of elements using Bubble Sort. 3. Write a JavaScript program to calculate grades of students and average grade of a class (use <code>prompt</code> to get input).
5	Form Validation using JavaScript: Write the client-side scripts for validating an HTML registration form (form validation) using JavaScript.
6	PHP and MySQL Integration: Develop a PHP application that performs the following operations: • Connect to a MySQL database server • Insert user data (name, email) through an HTML form • Retrieve and display all records from the database • Handle database connection and query errors appropriately
7	Familiarize Server Side Scripting using Node.js: Write a Node.js program to implement a simple HTTP web server that responds with different content based on the URL path requested. The server should handle at least three routes: • <code>/</code> – Respond with “Welcome to Node.js Web Server” • <code>/about</code> – Respond with a brief description of Node.js • <code>/contact</code> – Respond with contact information in HTML format

8	Programs using MongoDB database with Node.js: Develop a Node.js application that performs basic CRUD operations on a MongoDB database for managing a library collection. The application should implement the following functionalities: 1. Insert a new book with fields: title, author, genre 2. Display all books 3. Update a book's genre using its title 4. Delete a book using its title
9	Programs using MongoDB database with Node.js and React: Design and develop a full-stack web application using React (frontend) and Node.js with Express and MongoDB (backend) to manage student records. The application must implement a RESTful API with the following endpoints: 1. POST /students → Insert a new student (name, email, course) 2. GET /students → Retrieve all students 3. PUT /students/:email → Update a student's course using email 4. DELETE /students/:email → Delete a student using email The React frontend should: 1. Provide a form to add a new student 2. Display all student records 3. Allow updating and deleting students using the REST API
10	Develop a web application with HTML, CSS, JavaScript, PHP and MySQL (Groupwise projects)
11	Develop a web application with HTML, CSS, JavaScript, Node.js and MongoDB (Groupwise projects)

Reference Books

1. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, "Internet and World Wide Web How To Program", 5/E, Pearson Education, 2018.
2. Jon Duckett , "HTML and CSS: Design and Build Websites", Wiley, 1st Edition 2011.
3. Jon Duckett , "JavaScript and JQuery : Interactive Front-End Web Development", Wiley, 1st Edition 2013.
4. P.J. Deitel & H.M. Deitel, "Internet and World Wide Web How to program", 4th Edition, Pearson Education.
5. HTML 5 Black Book, Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP and jQuery, 2/e, 2016.

B24CSM51	DATABASE PROGRAMMING (Minor)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		2024

Preamble

This course introduces the basic ideas of databases and how to work with them using SQL and PL/SQL. Students learn how to create tables, write queries, use transactions, and develop stored procedures, functions, and triggers. The course focuses on hands-on practice so that students can build simple applications that use a database.

Prerequisites

Basic programming knowledge (variables, loops, conditions). Familiarity with any programming language.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain relational database concepts, SQL commands, and PL/SQL structures (Cognitive Knowledge Level: Understand)
CO 2	Apply SQL queries and PL/SQL blocks for data retrieval, modification, and transaction handling (Cognitive Knowledge Level: Apply)
CO 3	Develop practical solutions for database problems using SQL and PL/SQL (Cognitive Knowledge Level: Apply)
CO 4	Implement stored procedures, functions, and triggers for common application scenarios (Cognitive Knowledge Level: Apply)
CO 5	Build simple database-backed mini-projects demonstrating CRUD, transactions, and exception handling (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2										
CO 2	2	3	2	3	3	2		2	3	3	
CO 3	2	3	3	3	3	3		3	3	3	3
CO 4	2	3	3	3	3	3		3	3	3	3
CO 5	2	3	3	3	3	3	3	3	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	50	50	50
Apply	20	20	20
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (8 hours)

Introduction to Databases & SQL Basics:

Databases – DBMS – Need for structured data – Relational model – Tables, attributes, tuples – Keys and constraints – SQL basics: CREATE TABLE, INSERT, UPDATE, DELETE – SELECT with WHERE – ORDER BY – simple JOINS.

MODULE 2 (9 hours)

Relational Design & Intermediate SQL:

ER diagrams – Mapping ER to relational schema – Functional dependencies – Normalization up to 3NF (concept + examples) – SQL subqueries – GROUP BY – HAVING – Set operations (UNION, INTERSECT, MINUS) – Views – SQL exercises.

MODULE 3 (8 hours)

Advanced SQL for Transaction Processing:

Transactions and ACID – COMMIT / ROLLBACK – Isolation levels (concept) – Indexing and query performance basics – Using EXPLAIN – Database security basics: GRANT / REVOKE – Worked SQL examples.

MODULE 4 (9 hours)

PL/SQL Fundamentals and Stored Code:

PL/SQL architecture – Variables and data types – Input/output using substitution variables – Conditional and loop constructs – Cursors (implicit & explicit) – Stored procedures – Stored functions – Exception handling – Practical PL/SQL blocks.

MODULE 5 (11 hours)

Advanced PL/SQL and Database Applications:

Triggers: BEFORE/AFTER, row-level & statement-level – Packages (specification & body overview) – Dynamic SQL basics – Transactions in PL/SQL – Case studies: student record system, payroll, inventory – Mini-project guidelines.

Text Books

1. Silberschatz, Korth, Sudarshan — Database System Concepts, 7/e, McGraw-Hill
2. Ivan Bayross — SQL, PL/SQL: The Programming Language of Oracle, BPB Publications

Reference Books

1. Steven Feuerstein — Oracle PL/SQL Programming, 6/e, O'Reilly

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction to Databases & SQL Basics		8 hours
1.1	Introduction to DBMS and databases	1
1.2	Relational model, tables, attributes, tuples	1
1.3	Keys and integrity constraints	1
1.4	CREATE, INSERT, UPDATE, DELETE statements	1
1.5	SELECT with WHERE clause	1
1.6	ORDER BY, simple filtering	1
1.7	Simple JOIN operations	1
1.8	Lab exercises on basic SQL queries	1
Module 2: Relational Design & Intermediate SQL		9 hours
2.1	ER diagrams and relational mapping	1
2.2	Functional dependencies and 1NF/2NF/3NF (concept)	1
2.3	SQL subqueries	2
2.4	JOINS: inner, left, right, full	2
2.5	GROUP BY and HAVING clauses	1
2.6	Set operations (UNION, INTERSECT, MINUS)	1
2.7	Views creation and usage	1
2.8	Lab exercises: intermediate SQL queries	1

Module 3: Advanced SQL for Transaction Processing		8 hours
3.1	Transactions and ACID properties	1
3.2	COMMIT and ROLLBACK examples	1
3.3	Isolation levels (concept)	1
3.4	Indexing and query performance basics	1
3.5	Using EXPLAIN plans	1
3.6	Database security basics: GRANT / REVOKE	1
3.7	Error handling in SQL	1
3.8	Lab exercises: transaction and security commands	1
Module 4: PL/SQL Fundamentals and Stored Code		9 hours
4.1	PL/SQL architecture and block structure	1
4.2	Variables, constants, and data types	1
4.3	Conditional statements (IF-ELSE)	1
4.4	Loop constructs (FOR, WHILE, nested loops)	1
4.5	Cursors: implicit and explicit	2
4.6	Stored procedures	1
4.7	Stored functions	1
4.8	Exception handling and lab exercises	1
Module 5: Advanced PL/SQL and Database Applications		11 hours
5.1	Triggers: BEFORE/AFTER, row-level & statement-level	2
5.2	Packages: specification and body overview	1
5.3	Dynamic SQL (concept and simple example)	1
5.4	Transactions in PL/SQL	1
5.5	Case study: student record system	2
5.6	Case study: payroll system	1
5.7	Case study: inventory system	1
5.8	Mini-project guidelines and lab implementation	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the need for a database management system. Compare files and databases with suitable examples.
2. Using an example, demonstrate the use of basic SQL commands such as CREATE, INSERT, UPDATE, DELETE, and SELECT.

Course Outcome 2 (CO2):

1. Draw an ER diagram for a given scenario and convert it into a relational schema.
2. Explain functional dependencies and normalization up to 3NF with examples.

Course Outcome 3 (CO3):

1. Define transactions and ACID properties. Explain how isolation levels affect concurrent execution.
2. Demonstrate the use of GRANT and REVOKE in database security with an example.

Course Outcome 4 (CO4):

1. Write PL/SQL programs to demonstrate conditional statements, loops, and cursor operations.
2. Explain the structure of a stored procedure and stored function with examples.

Course Outcome 5 (CO5):

1. Develop a PL/SQL trigger to enforce a business rule using BEFORE or AFTER events.
2. Explain the concept of packages and dynamic SQL. Provide an example of a scenario where dynamic SQL is required.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CSM51

Course Name: DATABASE PROGRAMMING (Minor)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define DBMS. Give any three advantages of using a DBMS.
2. Compute the output of the following SQL query:

`SELECT AVG(Salary) FROM Employee WHERE Dept='HR';`

3. Explain primary key, candidate key, and foreign key with examples.
4. Consider the relation STUDENT(RollNo, Name, Dept, CGPA). Write the relational algebra expression to find all CSE students with CGPA ≥ 8 .
5. Write the syntax of a nested subquery and give one example.
6. Differentiate between INNER JOIN and FULL OUTER JOIN.
7. What are ACID properties? Briefly explain any two.
8. Write a PL/SQL block to print "Welcome to PL/SQL".
9. What is the purpose of a cursor? Mention the two types.
10. Write any three differences between stored procedures and stored functions.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the three-schema architecture of DBMS with a neat diagram. (7)
(b) Discuss the relational model concepts: relation, tuple, attribute, and domain. (7)

OR

12. (a) What are integrity constraints? Explain primary key, foreign key, and check constraint with SQL examples. (7)
(b) Write and explain SQL commands for CREATE, INSERT, UPDATE, and DELETE with examples. (7)
13. (a) Explain 1NF, 2NF, and 3NF. Normalize the given relation appropriately: R(A,B,C,D) with FDs: $A \rightarrow B$, $B \rightarrow C$, $C \rightarrow D$. (7)
(b) Write SQL queries using GROUP BY and HAVING for the EMPLOYEE table. (7)

OR

14. (a) Explain different types of JOINS with examples. (7)
(b) Write SQL queries demonstrating UNION, INTERSECT, and MINUS using suitable tables. (7)
15. (a) Explain ACID properties in detail with examples. (7)
(b) What are isolation levels? Explain READ COMMITTED and SERIALIZABLE. (7)

OR

16. (a) Explain indexing. Discuss B+ tree indexing with a diagram. (7)
(b) Explain GRANT and REVOKE with suitable SQL code. (7)
17. (a) Explain the structure of a PL/SQL block with example. (7)
(b) Write PL/SQL programs to demonstrate IF-ELSE and LOOP constructs. (7)

OR

18. (a) Explain explicit cursors with an example PL/SQL program. (7)
(b) Write short notes on stored procedures and stored functions. (7)
19. (a) Explain BEFORE and AFTER triggers with syntax and examples. (7)
(b) Write a PL/SQL program to create a trigger that logs salary updates. (7)

OR

20. (a) Explain package specification and package body with examples. (7)
(b) Write short notes on dynamic SQL and mini-project design guidelines. (7)

B24CSM52	CONCEPTS IN DEEP LEARNING (Minor)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		2024

Preamble

This course aims to introduce the learner to an overview of the concepts and algorithms involved in deep learning. Basic concepts and application areas of machine learning, deep networks, convolutional neural network and recurrent neural network are covered in this course. This is a foundational program that will help students understand the capabilities, challenges, and consequences of deep learning and prepare them to participate in the development of leading-edge AI technology. They will be able to gain the knowledge needed to take a definitive step in the world of AI.

Prerequisites

Basics of linear algebra and probability.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Demonstrate basic concepts in machine learning. (Cognitive knowledge: Understand)
CO 2	Illustrate the validation process of machine learning models using hyperparameters and validation sets. (Cognitive knowledge: Understand)
CO 3	Demonstrate the concept of the feed-forward neural network and its training process. (Cognitive knowledge: Apply)
CO 4	Build CNN and Recurrent Neural Network (RNN) models for different use cases. (Cognitive knowledge: Apply)
CO 5	Use different neural network/deep learning models for practical applications. (Cognitive knowledge: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	1				1	1				2
CO 2	2	3	2	3	3						1
CO 3	2	3	2	3	3	1	1	1	3	2	2
CO 4	2	3	2	3	3	1	2	1	3	2	3
CO 5	2	2	1	3	3	3	2	1	2		3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Introduction

Key components - Data, models, objective functions, optimization algorithms, Learning algorithms. Supervised learning- regression, classification, tagging, web search, page ranking, recommender systems, sequence learning, Unsupervised learning, Reinforcement learning, Historical Trends in Deep Learning. Other Concepts - overfitting, underfitting , hyperparameters and validation sets, estimators, bias and variance.

MODULE 2 (10 hours)

Optimization and Neural Networks

Neural Networks –Perceptron, Gradient Descent solution for Perceptron, Multilayer perceptron, activation functions, architecture design, chain rule, back propagation, gradient based learning. Introduction to optimization– Gradient based optimization, linear least squares. Stochastic gradient descent, Building ML algorithms and challenges.

MODULE 3 (9 hours)

Convolutional Neural Network

Convolutional Neural Networks – convolution operation, motivation, pooling, Convolution and Pooling as an infinitely strong prior, variants of convolution functions, structured outputs, data types, efficient convolution algorithms.

MODULE 4 (10 hours)

Recurrent Neural Network

Recurrent neural networks – Computational graphs, RNN design, encoder – decoder sequence to sequence architectures, deep recurrent networks, recursive neural networks, modern RNNs LSTM and GRU, Practical use cases for RNNs.

MODULE 5 (7 hours)

Application Areas

Applications – computer vision, speech recognition, natural language processing. Research Areas – Autoencoders, Representation learning, Boltzmann Machines, Deep belief networks.

Text Books

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press 2015 ed.
2. Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola, Dive into Deep Learning, August 2019.
3. Neural Networks and Deep Learning: A Textbook by Charu C. Aggarwal. Springer.1st edition, 2018.

Reference Books

1. Neural Smithing: Supervised Learning in Feed forward Artificial Neural Networks by Russell Reed, Robert J MarksII, 1st edition, 1999, MIT Press.
2. Practical Convolutional Neural Networks by Mohit Sewak, Md. Rezaul Karim, Pradeep Pujari, 1st edition, 2018, Packt Publishing Ltd.
3. Hands-On Deep Learning Algorithms with Python by Sudharsan Ravichandran, 1st edition, 2019, Packt Publishing Ltd.
4. Deep Learning with Python by Francois Chollet, 2nd edition, 2018, Manning Publications Co.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction		9 hours
1.1	Key components - Data, models, objective functions, optimization algorithms. (TB2: Section 1.1-1.2)	1
1.2	Learning algorithm (TB1: Section 5.1), Supervised learning- regression, classification, tagging, web search, page ranking (TB2: Section 1.3.1)	1
1.3	Recommender systems, Sequence learning, Unsupervised learning, Reinforcement learning(TB2: Section 1.3.2-1.3.4)	1
1.4	Historical Trends in Deep Learning (TB1: Section 1.2).	1
1.5	Concepts: overfit, underfit, hyperparameters and validation sets. (TB1: Section 5.2-5.3)	1

1.6	Concepts: Estimators, bias and variance. (TB1: Section 5.4)	1
1.7	Demonstrate the concepts of supervised learning algorithms using a suitable platform.	2
1.8	Demonstrate the concepts of unsupervised using a suitable platform.	1
Module 2: Optimization and Neural Networks		10 hours
2.1	Perceptron, Stochastic Gradient descent, Gradient descent solution for perceptron (TB3: Section 1.1 - 1.2.1)	1
2.2	Multilayer perceptron (TB3: Section 1.2.2), (TB1: Section 6.1,6.3)	1
2.3	Activation functions- Sigmoid, tanh, Softmax, ReLU, leaky ReLU (TB3: Section 1.2.1.3 - 1.2.1.5)	1
2.4	Architecture design (TB1: Section 6.4, TB3: Section 1.6)	1
2.5	Chain rule, back propagation (TB3: Section 1.3)	1
2.6	Gradient based learning (TB1: Section 6.2)	2
2.7	Gradient based optimization (TB1: Section 4.3)	1
2.8	Linear least squares using a suitable platform. (TB1: Section 4.5)	1
2.9	Building ML Algorithms and Challenges (TB3: 1.4, TB1: 5.10-5.11)	1
Module 3: Convolution Neural Network		9 hours
3.1	Convolution operation (TB1:Section 9.1)	1
3.2	Motivation, pooling (TB1:Section 9.2-9.3)	1
3.3	Convolution and Pooling as an infinitely strong prior (TB1: Section 9.4)	1
3.4	Variants of convolution functions – multilayer convolutional network, tensors, kernel flipping, downsampling, strides and zero padding. (TB1: Section 9.5)	2
3.5	Variants of convolution functions - unshared convolutions, tiled convolution, training different networks. (TB1: Section 9.5)	1
3.6	Structured outputs, data types (TB1: Section 9.6-9.7)	1
3.7	Efficient convolution algorithms. (TB1: Section 9.8,9.10)	1

3.8	Case Study: AlexNet, VGG, ResNet. (TB3: Section 8.4.1, 8.4.3, 8.4.5)	1
Module 4: Recurrent Neural Network		10 hours
4.1	Computational graphs (TB1: Section 10.1)	2
4.2	RNN (TB1: Section 10.2-10.3)	1
4.3	Encoder – decoder sequence to sequence architectures. (TB1: Section 10.4)	1
4.4	Deep recurrent networks (TB1: Section 10.5)	2
4.5	Recursive neural networks , Modern RNNs, LSTM and GRU (TB1: Section 10.6, 10.10)	2
4.6	Practical use cases for RNNs. (TB1: Section 11.1-11.4)	1
4.7	Demonstrate the concepts of RNN using a suitable platform.	1
Module 5: Applications and Research		7 hours
5.1	Computer vision. (TB1: Section 12.2)	2
5.2	Speech recognition. (TB1: Section 12.3)	1
5.3	Natural language processing. (TB1: Section 12.4)	2
5.4	Brief introduction on current research areas- Autoencoders, Representation learning. (TB1: Section 14.1-14.2, TB3: 9.3)	1
5.5	Brief introduction on current research areas- Boltzmann Machines, Deep belief networks. (TB1: Section 20.1, 20.3)	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Compare regression and classification.
2. Define supervised learning? Distinguish between regression and classification.
3. Discuss the different learning approaches used in machine learning.
4. You train an initial model that achieves a 90kind of problems your model is experiencing, and suggest a possible solution.

5. How does splitting a dataset into train, validation and test sets help identify overfitting?
6. Consider solving a classification task. You first train your network on 20 samples. Training converges, but the training loss is very high. You then decide to train this network on 10,000 examples. Is your approach to fixing the problem correct? If yes, explain the most likely results of training with 10,000 examples. If not, give a solution to this problem.

Course Outcome 2 (CO 2):

1. What are hyperparameters? Why are they needed?
2. What issues are to be considered while selecting a model for applying machine learning in a given problem?
3. Update the parameters V_{11} in the given MLP using back propagation with learning rate as 0.5 and activation function as sigmoid. Initial weights are given as $V_{11}=0.2$, $V_{12}=0.1$, $V_{21}=0.1$, $V_{22}=0.3$, $V_{11}=0.2$, $W_{11}=0.5$, $W_{21}=0.2$
4. Draw the architecture of a multi-layer perceptron.
5. Derive update rules for parameters in the multi-layer neural network through the gradient descent.
6. Why is it important to place non-linearities between the layers of neural networks?
7. You design a fully connected neural network architecture where all activations are sigmoids. You initialize the weights with large positive numbers. Is this a good idea? Explain your answer.

Course Outcome 3 (CO 3):

1. Give two benefits of using convolutional layers instead of fully connected ones for visual tasks.
2. Suppose that a CNN was trained to classify images into different categories. It performed well on a validation set that was taken from the same source as the training set but not on a testing set. What could be the problem with the training of such a CNN? How will you ascertain the problem? How can those problems be solved?
3. You are given a dataset of 10×10 grayscale images. Your goal is to build a 5-class classifier. You have to adopt one of the following two options: a) the input is flattened into a 100-dimensional vector, followed by a fully-connected layer with 5 neurons, b) the input is directly given to a convolutional layer with five 10×10 filters. Explain which one you would choose and why.
4. Weight sharing allows CNNs to deal with image data without using too many parameters. Does weight sharing increase the bias or the variance of a model?

5. Why do the layers in a deep architecture need to be non-linear?
6. A convolutional neural network has 4 consecutive layers as follows: 3 x 3 conv (stride 2) - 2 x 2 Pool - 3 x 3 conv (stride 2) - 2 x 2 Pool How large is the set of image pixels which activate a neuron in the 4th non-image layer of this network?

Course Outcome 4 (CO 4):

1. Explain how the cell state is updated in the LSTM model from C_{t-1} to C_t
2. Show the steps involved in an LSTM to predict stock prices.
3. Illustrate the workings of the RNN with an example of a single sequence defined on a vocabulary of four words.
4. If we have a recurrent neural network (RNN), we can view it as a different type of network by "unrolling it through time". Briefly explain what that means.
5. Briefly explain how "unrolling through time" is related to "weight sharing" in convolutional networks.
6. Explain how the cell state is updated in the LSTM model from C_{t-1} to C_t
7. Show the steps involved in an LSTM to predict stock prices. Give one advantage of using an RNN rather than a convolutional network.

Course Outcome 5 (CO 5):

1. Development a deep learning solution for problems in the domain i) natural language processing or ii Computer vision.
2. Is an autoencoder for supervised learning or for unsupervised learning? Explain briefly.
3. Sketch the architecture of an autoencoder network.
4. Describe how to train an autoencoder network.
5. Write down the formula for the energy function (E) of a Restricted Boltzmann Machine (RBM).

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CSM52

Course Name: CONCEPTS IN DEEP LEARNING (Minor)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Distinguish between supervised learning and Reinforcement learning. Illustrate with an example.
2. Differentiate classification and regression.
3. Compare overfitting and underfitting. How it can affect model generalization.
4. Why does a single perceptron cannot simulate simple XOR function? Explain how this limitation is overcome?
5. Illustrate the strengths and weaknesses of convolutional neural networks.
6. Illustrate convolution and pooling operation with an example
7. How many parameters are there in AlexNet? Why the dataset size (1.2 million) is important for the success of AlexNet?
8. Explain your understanding of unfolding a recursive or recurrent computation into a computational graph.
9. Illustrate the use of deep learning concepts in Speech Recognition.
10. What is an autoencoder? Give one application of an autoencoder

PART B

Answer any one question from each module. Each question carries 14 marks.

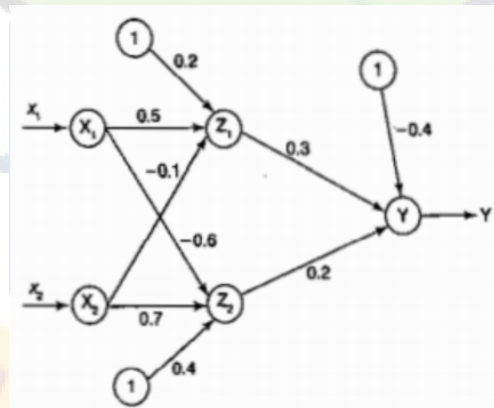
11. (a) “A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P , if its performance at tasks in T , as measured by P , improves with experience E .” What is your understanding of the terms task, performance and experience. Explain with two examples (10)
- (b) How does bias and variance trade-off affect machine learning algorithms? (4)

OR

12. (a) Illustrate the concepts of Web search, Page Ranking, Recommender System with suitable examples. (10)
- (b) List and discuss the different hyper parameters used in fine tuning the traditional machine learning models (4)
13. (a) How multilayer neural networks learn and encode higher level features from input features. (7)
- (b) Explain gradient descent and delta rule? Why stochastic approximation to gradient descent is needed? (7)

OR

14. (a) Find the new weights for the network using backpropagation algorithm, the network is given with an input pattern $[-1, 1]$ and target output as $+1$, Use learning rate of $\alpha=0.3$ and bipolar sigmoid function. (7)



- (b) Write an algorithm for backpropagation which uses stochastic gradient descent method. Comment on the effect of adding momentum to the network. (7)
15. (a) Input to CNN architecture is a color image of size $112 \times 112 \times 3$. The first convolution layer comprises of 64 kernels of size 5×5 applied with a stride of 2 and padding 0. What will be the number of parameters? (5)
- (b) Let $X = [-1, 0, 3, 5]$ $W = [0.3, 0.5, 0.2, 1]$ be the input of i th layer of a neural network and to apply softmax function. What should be the output of it? (4)
- (c) Draw and explain the architecture of convolutional network (5)

OR

16. (a) Explain the concept behind i) Early stopping ii) dropout iii) weight decay (9)
(b) How backpropagation is used to learn higher-order features in a convolutional Network? (5)
17. (a) Explain the working of RNN and discuss how backpropagation through time is used in recurrent networks. (8)
(b) Describe the working of a long short term memory in RNNs. (6)

OR

18. (a) What is the vanishing gradient problem and exploding gradient problem? (8)
(b) Why do RNNs have a tendency to suffer from exploding/vanishing gradient? How to overcome this challenge? (6)
19. (a) Explain any two word embedding techniques. (8)
(b) Explain the merits and demerits of using Auto encoders in Computer Vision. (6)

OR

20. (a) Illustrate the use of representation learning in object classification. (7)
(b) Compare Boltzmann Machine with Deep Belief Network. (7)

B24CSM53	CLIENT SERVER SYSTEMS (Minor)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

Client-Server Systems introduces the concepts, architecture, and components of client-server computing. The course covers the evolution of client-server systems, client and server hardware and software, standards, and implementation challenges. It also examines web-based client-server models, databases, middleware, and enterprise applications. The subject enables students to understand and apply client-server concepts in modern distributed systems.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Describe the fundamentals, evolution, and application areas of client-server computing, including trends in hardware, software, networking, and business environments (Cognitive Knowledge Level: Remember)
CO 2	Outline the Classifications and architectures of client-server Systems (Cognitive Knowledge Level: Understand)
CO 3	Apply client-side concepts related to hardware, operating systems, graphical interfaces, database access, and software products to identify client requirements (Cognitive Knowledge Level: Apply)
CO 4	Apply server-side concepts including server architecture, operating systems, transaction processing, data management, and reliability mechanisms in client-server environments (Cognitive Knowledge Level: Apply)
CO 5	Apply client-server concepts to web-based and enterprise systems using web architectures, databases, middleware, data warehouses, and groupware technologies (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2			1		1			1	2	2
CO 2	1	2	2	2	1	2			3	1	2
CO 3	1	2	3	2	3		2		2	2	3
CO 4	2	3	3	3	3	3	2		3	3	3
CO 5	2	3	3	3	3	3	3	3	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction to Client Server Computing:

Overview of Client Server Computing: What is Client Server Computing, Benefits of Client Server Computing. Evolution of client server computing: Hardware Trends, Software Trends, Evolution of Operating Systems, Networking Trends, Business Considerations. Client/Server Development Tools, Development of Client/Server Systems, Single System Image, Downsizing and Rightsizing, Overview of Client Server Applications: Components of Client/Server Applications, Classes of Client/Server Applications, Categories of Client/Server Applications.

MODULE 2 (7 hours)

Client–Server Application Components :

Classification of Client/server Systems- Two-Tier Computing, Middleware, Three-Tier Computing - Model View Controller (MVC), Principles behind Client/ Server Systems, Client/Server Topologies, Existing Client/Server Architecture, Architecture for Business Information System.

MODULE 3 (8 hours)

The Client:

Client - Client components, Client Operating Systems, GUI, X windows vs. Windowing, Database Access, Application Logic. Services, Request for services, RPC, Windows Services, Print Services, Remote Boot Services, Other remote services, Utility services, Dynamic Data Exchange (DDE), Object Linking and Embedding (OLE), Common Object Request Broker Architecture (CORBA). Client Software Products, Client Requirements.

MODULE 4 (11 hours)

The Server:

Server Hardware: Categories of Servers, Features of Server Machines, classes of Server

Machines. Server Environment: Layers of Software, Network management environment, Network Computing Environment. Server Operating System. Server requirements: Platform Independence, Transaction Processing, Connectivity, Intelligent Database, Stored Procedure, Triggers, Load Levelling, Optimizer, Testing and Diagnostic Tools, Reliability, Backup and Recovery Mechanism. Server Data Management and Access Tools: Data Manager Features, Data Management Software, Database Gateways.

MODULE 5 (12 hours)

Client Server and Internet:

Web client/server - Client/Server Web Style, URL, HTML, HTTP- 3 tier client/server, web style, CGI The server side of web - CGI Scenario, CGI and State. SQL Database Servers, SQL Middleware and federated databases, Data Warehouses, EIS/DSS to data mining, Client/Server GroupWare Server - what is GroupWare, Components of GroupWare.

Text Books

1. Dawana Travis Dewire, "Client Server Computing", Tata Mc-Graw Hill Education Pvt.Ltd., New Delhi, 2003
2. Robert Orfali, Dan Harkey & Jeri Edwards, "Essential Client/Server Survival Guide", second edition, John Wiley & Sons, Singapore, 2003.

Reference Books

1. Eric J Johnson, "A complete guide to Client / Server Computing", first edition, Prentice Hall, New Delhi, 2001.
2. Smith & Guengerich, "Client /Server Computing", Prentice Hall, New Delhi, 2002
3. James E. Goldman, Phillip T. Rawles, Julie R. Mariga, "Client/Server Information Systems, A Business Oriented Approach", John Wiley & Sons, Singapore, 2000.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction to Client Server Computing		7 hours
1.1	Overview of Client Server Computing: What is Client Server Computing	1
1.2	Benefits of Client Server Computing	1

1.3	Evolution of client server computing: Hardware Trends, Software Trends, Evolution of Operating Systems, Networking Trends, Business Considerations.	1
1.4	Client/Server Development Tools, Development of Client/Server Systems	1
1.5	Overview of Client Server Applications: Components of Client/Server Applications,	1
1.6	Single System Image, Downsizing and Rightsizing	1
1.7	Classes of Client/Server Applications, Categories of Client/Server Applications	1
Module 2: Understanding Client–Server Computing		7 hours
2.1	Classification of Client/server Systems, Open System Standards	1
2.2	Two-Tier Computing,	1
2.3	Middleware, Three-Tier Computing	1
2.4	Model View Controller (MVC)	1
2.5	Client/Server Topologies,	1
2.6	Existing Client/Server Architecture,	1
2.7	Architecture for Business Information System.	1
Module 3: The Client		8 hours
3.1	Client - Client components, Client Operating Systems, GUI, X windows vs. Windowing	1
3.2	Database Access, Application Logic.	1
3.3	Services, Request for services, RPC, Windows Services	1
3.4	Print Services, Remote Boot Services, Other remote services, Utility services	1
3.5	Dynamic Data Exchange (DDE), Object Linking and Embedding (OLE)	1
3.6	Common Object Request Broker Architecture (CORBA)	1
3.7	Client Software Products, Client Requirements: GUI design standards, Open GUI Standards	1
3.8	Interface Independence, Testing Interfaces, Development Aids	1
Module 4: The Server		11 hours
4.1	Server Hardware: Categories of Servers	1

4.2	Features of Server Machines,	1
4.3	Classes of Server Machines.	1
4.4	Server Environment: Layers of Software, Network management environment, Network Computing Environment	1
4.5	Server Operating System.	1
4.6	Server requirements: Platform Independence, Transaction Processing	1
4.7	Connectivity, Intelligent Database, Stored Procedure	1
4.8	Triggers, Load Levelling, Optimizer, Testing and Diagnostic Tools, Reliability	1
4.9	Backup and Recovery Mechanism.	1
4.10	Server Data Management and Access Tools: Data Manager Features	1
4.11	Data Management Software, Database Gateways.	1
Module 5: Client Server and Internet		12 hours
5.1	Web client/server - Client/Server Web Style	1
5.2	URL, HTML	1
5.3	HTTP	1
5.4	3 tier client/server, web style	1
5.5	CGI The server side of web - CGI Scenario, CGI and State.	1
5.6	SQL Database Servers: Fundamentals of SQL and Relational Databases	1
5.7	SQL Database Server Architectures	1
5.8	Stored Procedure, Triggers and Rules	1
5.9	SQL Middleware and federated database - SQL Middleware: The Options	1
5.10	Data Warehouses: The Data Warehouse	1
5.11	EIS/DSS to data mining	1
5.12	Client/Server GroupWare Server - what is GroupWare, Components of GroupWare	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the driving forces behind the development of client - server systems from different perspectives.

Course Outcome 2 (CO 2):

1. Explain how two-tier and three-tier architectures enhance the performance of client-server systems.

Course Outcome 3 (CO 3):

1. Explain the role of client in client/server computing and also explain the various services provided by the client.
2. What is the primary motivation behind the RPC facility? How does RPC facility simplify the job of Distributed application Programmers?

Course Outcome 4 (CO 4):

1. Describe the role of servers in client-server systems and explain how server requirements and data management tools improve performance and reliability.

Course Outcome 5 (CO 5):

1. Explain the web client/server architecture in detail. Discuss URL, HTML, HTTP, and CGI with state management.
2. Explain SQL database servers, SQL middleware, federated databases, and data warehouses in a client/server environment.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CSM53

Course Name: CLIENT SERVER SYSTEMS (Minor)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. How Client/Server Computing environment is different from Mainframe based computing environment?
2. Write short notes on Single System Image and Downsizing in client-server systems.
3. List the components of MVC architecture.
4. Discuss the relevance of Clients/Server system in adopting open system standards. Justify your answer.
5. Enumerate the services provided in a client/server system.
6. List out the features of network operating system.
7. What is meant by platform independence in servers?
8. What is the role of a database gateway?
9. Apply the concept of CGI to explain how a web form submission is processed at the server side.
10. Apply the concept of CGI to explain how a web form submission is processed at the server side.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the driving forces behind the development of client - server systems from different perspectives. (10)
(b) Discuss the benefits and limitations of client-server computing. (4)

OR

12. (a) Explain client-server computing in detail. Discuss its characteristics, architectures, and advantages with suitable diagrams. (14)
13. (a) Explain how two-tier and three-tier computing improve the performance of client-server systems (10)
(b) Discuss the principles behind client-server systems. (4)

OR

14. (a) Explain the architecture of a business information system. (8)
(b) Discuss the various techniques used to improve the performance of client-server applications. (6)
15. (a) Explain the architecture of CORBA (8)
(b) Describe the service request mechanism in client-server systems. (6)

OR

16. (a) In Client/Server computing, explain the following with example in detail (a) Dynamic Data Exchange (b) RPC (c) Remote Boot Service (d) Object- linking and embedding. (10)
(b) Explain the role of client in Client/Server computing and also explain the various services provide by client. (4)
17. (a) Describe the layers of software in a server environment. (7)
(b) Explain the categories and classes of server machines. (7)

OR

18. (a) Describe the role of servers in client-server systems and explain how server requirements and data management tools improve performance and reliability. (14)
19. (a) Apply the concept of data warehousing to explain how organizational data can be transformed for decision support. (7)
(b) Explain how GroupWare components can be applied to support collaboration in a distributed software development team. (7)

OR

20. (a) Explain the web client/server architecture in detail. Discuss URL, HTML, HTTP, and CGI with state management. (14)

B24CSH51	ADVANCED CONCEPTS IN COMPUTER VISION (Honours)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

Computer vision is a field of artificial intelligence (AI) that enables computers and systems to derive meaningful information from digital images, videos and other visual inputs. The curriculum covers the basics of image formation, key computer vision concepts, methods, techniques, pattern recognition, various problems in designing computer vision, object recognition systems and image based modeling and rendering. This course enables the learners to understand the fundamentals of computer vision and develop applications in computer vision.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Summarize basic concepts, terminology, theories, models and methods in the field of computer vision. (Cognitive Knowledge Level: Understand)
CO 2	Explain basic methods of computer vision related to multi-scale representation, edge detection, detection of other primitives, stereo, motion and object recognition. (Cognitive Knowledge Level: Apply)
CO 3	Describe principles of Segmentation, Motion Segmentation and Classification (Cognitive Knowledge Level: Apply)
CO 4	Apply image restoration techniques to identify and remove various types of noise (periodic, salt-and-pepper) from images. (Cognitive Level: Apply)
CO 5	Demonstrate an understanding of object detection and recognition strategies and their implementation in real-world scenarios using tools like MediaPipe. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1	1	1	1	3	2	1	1	1	2
CO 2	2	1	1	1	1	1		2	3	1	1
CO 3	2	1	1	1	1	1	1	2	3	1	1
CO 4	2	2	1	3	1	2	2	2	2	1	1
CO 5	2	2	2	3	3	3	3	3	3	3	2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	50	50	50
Apply	20	20	20
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (8 hours)

Image Formation and Filtering

Geometric Camera Models - Pinhole perspective, Intrinsic and Extrinsic Parameters, Geometric Camera Calibration. Linear Filters- Linear Filters and Convolution, Shift Invariant Linear Systems. Filters as Templates - Normalized Correlation and Finding Patterns.

MODULE 2 (9 hours)

Local Image Features and Stereo Vision

Image Gradients - Computing the Image Gradient, Gradient Based Edge and Corner Detection. Stereopsis- Binocular Camera Geometry, Epipolar Constraint, Binocular Reconstruction, Local Methods for Binocular Fusion, Global Methods for Binocular Fusion.

MODULE 3 (9 hours)

Segmentation

Segmentation - Background subtraction, Interactive segmentation, Forming image regions. Segmentation by clustering - Watershed Algorithm, Segmentation Using K-means. Fitting Lines and Planes-Fitting a single line, fitting a plane, fitting multiple lines. Motion Segmentation by Parameter Estimation- Optical Flow and Motion, Flow Models, Motion Segmentation with Layers.

MODULE 4 (9 hours)

Image Restoration and Compression

Image restoration: noise models, restoration in the presence of noise only, periodic noise reduction. Image compression: Basic compression methods: Huffman coding, arithmetic coding, LZW coding, run-length coding.

MODULE 5 (10 hours)

Finding Objects and Application

Object detection - The Sliding Window Method. Object Recognition -Goals of Object Recognition System.

Applications - Robot Navigation by stereo vision, Face detection, Face recognition, Activity

Recognition, Tracking people.
Familiarization of MediaPipe

Text Books

1. Forsyth, David, and Jean Ponce. Computer vision: A modern approach. Prentice hall, 2011.
2. Digital Image Processing (4 e) Rafael C. Gonzalez , Richard E. Woods

Reference Books

1. Szeliski, Richard, Computer vision: algorithms and applications. Springer Science & Business Media, 2010.
2. Medioni, Gerard, Emerging topics in computer vision. and Sing Bing Kang. Prentice Hall PTR, 2004.
3. Trucco, Emanuele, and Alessandro Verri, Introductory techniques for 3-D computer vision. Vol. 201. Englewood Cliffs: Prentice Hall, 1998.
4. Faugeras, Olivier, and Olivier Autor Faugeras, Three-dimensional computer vision: a geometric viewpoint. MIT press, 1993.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1:Image Formation and Filtering		8 hours
1.1	Geometric Camera model - Pinhole perspective	1
1.2	Geometric Camera model - Intrinsic Parameters	1
1.3	Geometric Camera model - Extrinsic Parameters	1
1.4	Geometric Camera Calibration – Linear Approach	1
1.5	Linear Filters and Convolution	1
1.6	Shift Invariant Linear Systems - Discrete convolution	1
1.7	Normalized Correlation	1

1.8	Finding patterns	1
Module 2: Local Image Features and Stereo Vision		9 hours
2.1	Local Image Features - Computing the Image Gradient	1
2.2	Gradient Based Edge Detection	1
2.3	Gradient Based Corner Detection	1
2.4	Stereopsis - Binocular Camera Geometry and Epipolar Constraint	1
2.5	Essential Matrix and Fundamental Matrix	1
2.6	Binocular Reconstruction	1
2.7	Local Methods for Binocular Fusion	1
2.8	Global Methods for Binocular Fusion(Lecture -1)	1
2.9	Global Methods for Binocular Fusion(Lecture -2)	1
Module 3: Segmentation		9 hours
3.1	Segmentation basics	1
3.2	Applications - Background Subtraction, Interactive Segmentation	1
3.3	Forming Image Regions	1
3.4	Segmentation by clustering - The Watershed Algorithm	1
3.5	Segmentation by clustering - Segmentation Using K-mean	1
3.6	Fitting a single line	1
3.7	fitting a plane,multiple line	1
3.8	Motion Segmentation by Parameter Estimation - Optical Flow and Motion	1
3.9	Flow Models and Motion Segmentation with Layers	1
Module 4: Image Restoration and Compression		9 hours
4.1	Image restoration- model of image restoration	1
4.2	Noise models- Gaussian model, Rayleigh model , Erlang model, Exponential model, uniform model and salt and pepper model	1
4.3	Restoration in the presence of noise only - Order-Statistics filters	1
4.4	Restoration in the presence of noise only - Adaptive Filters	1
4.5	Periodic noise reduction	1

4.6	Image compression- fundamentals	1
4.7	Basic compression methods: Huffman coding, arithmetic coding	1
4.8	LZW coding	1
4.9	Run-length coding	1
Module 5: Finding Objects and Image based modeling and Rendering		10 hours
5.1	Detecting Objects in Images- The Sliding Window Method -Face Detection	1
5.2	The Sliding Window Method- Detecting Humans	1
5.3	The Sliding Window Method- Detecting Humans- Detecting Boundaries	1
5.4	Object Recognition - Goals of Object Recognition System	1
5.5	Application of binocular stereo vision - Robot Navigation by stereo vision	1
5.6	Face detection	1
5.7	Face recognition	1
5.8	Activity recognition	1
5.9	Tracking people	1
5.10	Familiarization of MediaPipe	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. A robotic arm needs to pick up objects from a conveyor belt. Describe how understanding intrinsic and extrinsic camera parameters would be crucial for the robot to accurately locate and grasp the objects.
2. Imagine you are designing a surveillance system. How could a linear filter be used to detect sudden changes (like the appearance of a person) in a video feed, and what kind of filter would you choose?

Course Outcome 2 (CO 2):

1. A security camera system is designed to alert if a new object enters a restricted area. How can gradient-based edge detection be used to identify the boundaries of such an object, and why is this useful for an alert system?
2. You are building a system for a warehouse robot that needs to determine the depth of packages on shelves to avoid collisions. Explain how binocular camera geometry and the epipolar constraint enable the robot to achieve this 3D understanding.

Course Outcome 3 (CO 3):

1. You are developing an automated system to count specific types of cells in microscopic images. Explain how background subtraction could be used to isolate the cells from the slide, and what challenges might arise with varying illumination.
2. A user wants to quickly select a complex object (like a person) from a photograph to edit it separately. Describe how an interactive segmentation technique could be implemented, and why it might be preferred over a fully automated method in this scenario.

Course Outcome 4 (CO 4):

1. Given a specific degradation function $H(u, v)$ and an additive noise term $N(u, v)$ in the frequency domain, derive the formula for the restored image using the basic inverse filtering approach. Explain the primary practical challenge and a potential limitation of this method.
2. Demonstrate the step-by-step process of encoding the data sequence "BABAABABA" using the LZW algorithm, starting with an initial dictionary containing only 'A' (index 1) and 'B' (index 2).

Course Outcome 5 (CO 5):

1. A security system needs to detect unauthorized personnel in a restricted area. Describe how the "Sliding Window Method" would be applied to an incoming video stream to identify human figures, and discuss a major computational challenge this method might face.
2. You are developing an interactive museum exhibit where visitors can point their phone cameras at sculptures, and the app provides information about that specific piece. Explain how an "Object Recognition System" would achieve this, outlining its main goals and a key step in its process.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CSH51

***Course Name: ADVANCED CONCEPTS IN COMPUTER VISION
(Honours)***

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Explain the difference between intrinsic and extrinsic camera parameters. How does geometric camera calibration help account for both when a robot arm needs to precisely grasp an object?
2. Describe the core idea behind using "filters as templates" for finding patterns. If you were searching for a specific small icon on a webpage, how would normalized correlation improve the robustness of this search compared to simple convolution?
3. What does an "image gradient" represent, and how is it typically computed? In what way can understanding image gradients assist a smart surveillance system in detecting the presence of a new object entering a scene?
4. Illustrate epipolar geometry and show epipolar lines and epipoles.
5. Describe how "K-means clustering" can be adapted for image segmentation. If you're segmenting an image into regions of similar color, what would represent the 'features' for each pixel in the clustering process?
6. How does background subtraction help in segmenting an image?
7. Differentiate between Gaussian noise and Salt-and-pepper noise in terms of their origin, appearance in an image, and mathematical distribution characteristics.

8. Explain the core concept of LZW coding.
9. Explain the fundamental principle of the Sliding Window method in object detection. What problem does this systematic scanning approach aim to solve?
10. Differentiate between "Face Detection" and "Face Recognition." Explain what information is the output of each process.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Demonstrate the relationship between a point in the world coordinate frame and its corresponding image point using camera parameters. (8)
(b) Show that convolving a function with a delta function simply reproduces the original function. (6)

OR

12. (a) What is linear filtering? Explain two applications of linear filtering to image processing. (7)
(b) Explain an application of normalised correlation to find patterns. (7)
13. (a) Show that smoothing an image and then computing the gradient is same as convolving an image with the derivative of a smoothing function. (8)
(b) State the epipolar constraint and derive its representations using the Essential matrix and the Fundamental matrix. (6)

OR

14. (a) Explain the algorithm for computing edges using gradients. (8)
(b) Define binocular fusion. Explain two local methods for binocular fusion. (6)
15. (a) Discuss the different interactive segmentation approaches. (7)
(b) What is meant by optical flow? How can it be utilized for segmenting images? (7)

OR

16. (a) Explain the Watershed algorithm. (6)
(b) How can we perform motion segmentation by parameter estimation? (8)
17. (a) An image is corrupted by both Gaussian and impulse noise. Compare and contrast the effectiveness of the Arithmetic Mean filter and the Median filter in this scenario. Justify your selection of the optimal filter for a given application (7)

- (b) Explain the key differences in how Huffman coding (entropy-based) and LZW coding (dictionary-based) achieve compression. Discuss the trade-offs in terms of implementation complexity, need for prior information, and compression performance. (7)

OR

18. (a) An image contains large contiguous areas of a single color (e.g., a blue sky). Which basic compression method (Huffman, Arithmetic, LZW, RLE) would likely be most efficient in reducing redundancy for this specific image type, and why? (7)
- (b) You are designing a simple image compression pipeline. Propose a combination of two basic compression methods (e.g., RLE followed by Huffman coding, or similar) to maximize lossless compression efficiency for typical grayscale images. Justify the order and the rationale for each method in your pipeline. (7)
19. (a) Summarize how stereo vision is used in robot navigation systems to achieve depth perception. What information does the system extract from two separate camera views? (7)
- (b) Explain the steps for designing a complete face recognition system for a secure building entrance. (7)

OR

20. (a) A simple sliding window detector is deployed to find pedestrians in images taken from a fixed surveillance camera. The camera is always the same height above the ground. (7)
- (b) You are tasked with developing a "Face Recognition" system for secure building access, alongside a separate system for "Activity Recognition" to detect suspicious behaviors (e.g., loitering, unauthorized entry attempts). Discuss how these two application areas complement each other in enhancing overall security. Explain the technical challenges specific to each area and how they might be addressed. (7)

B24CSH52	MACHINE LEARNING IN COMPUTATIONAL BIOLOGY (Honours)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course introduces the integration of machine learning with computational biology, equipping students to analyze biological sequences and predict molecular structures. Students will master probabilistic models, neural networks, and algorithmic approaches tailored to genomic and proteomic data. The curriculum emphasizes both theoretical foundations and practical implementation for modern bioinformatics challenges.

Prerequisites

COMPUTATIONAL FUNDAMENTALS FOR BIOINFORMATICS

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Transform biological data into digital symbol sequences and apply probabilistic frameworks with Bayesian inference to model genomic diversity, protein structures, and molecular functions, including the underlying natural science principles and engineering fundamentals of biological systems.(Cognitive Knowledge Level: Apply)
CO 2	Implement core machine learning algorithms including dynamic programming, gradient descent, and evolutionary methods for omics data feature extraction and statistical mechanics applications, using mathematical and statistical concepts with appropriate engineering practice.(Cognitive Knowledge Level: Apply)
CO 3	Design neural network architectures for biological sequence analysis, implement backpropagation learning, and evaluate performance metrics for protein structure prediction and signal peptide identification, while considering resource efficiency and reproducibility in computational pipelines.(Cognitive Knowledge Level: Apply)
CO 4	Construct Hidden Markov Models with appropriate initialization and learning algorithms, apply them to protein and nucleic acid sequence analysis and structure prediction, and evaluate the societal and ethical implications (e.g., genomic privacy, bias in predictive models).(Cognitive Knowledge Level: Apply)
CO 5	Develop probabilistic graphical models, hybrid neural network architectures, and bidirectional recurrent networks for gene finding, DNA symmetry analysis, and advanced protein secondary structure prediction, demonstrating engagement with current research literature and inclusive team practices in bioinformatics projects.(Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1	1	1	1						1
CO 2	2	3	1	1	2		3				
CO 3	2	3	2	3	3	1					3
CO 4	2	2	1	3	2	2					1
CO 5				3	3	1	3				3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Foundations of Computational Biology:

Biological Data in Digital Symbol Sequences, Genomes—Diversity, Size, and Proteins and Proteomes, Prediction of Molecular Function and Structure, Machine-Learning Foundations: The Probabilistic Framework, Introduction: Bayesian Modeling, The Cox Jaynes Axioms, Bayesian Inference and Induction, Model Structures: Graphical Models and Other Tricks.

MODULE 2 (11 hours)

Probability and Information Theory

Probabilistic Modeling and Inference: Examples, The Simplest Sequence Models, Statistical Mechanics, Machine Learning Algorithms, Dynamic Programming, Gradient Descent, EM/GEM Algorithms, Markov-Chain Monte-Carlo Methods, Simulated Annealing, Evolutionary and Genetic Algorithms, Learning Algorithms: Miscellaneous Aspects.

MODULE 3 (10 hours)

Machine Learning Applications in Biology:

Neural Networks: The Theory, Universal Approximation Properties, Priors and Likelihoods, Learning Algorithms: Backpropagation, Neural Networks: Applications, Sequence Encoding and Output Interpretation, Sequence Correlations and Neural Networks, Prediction of Protein Secondary Structure, Prediction of Signal Peptides and Their Cleavage Sites, Applications for DNA and RNA Nucleotide Sequences, Prediction Performance Evaluation, Different Performance Measures,

MODULE 4 (8 hours)

Hidden Markov Models:

Hidden Markov Models: Prior Information and Initialization, Likelihood and Basic Algorithms, Learning Algorithms, Applications of HMMs: General Aspects, Hidden Markov

Models: Applications, Protein Applications, DNA and RNA Applications, Advantages and Limitations of HMMs.

MODULE 5 (7 hours)

Probabilistic Graphical Models in Bioinformatics

Probabilistic Graphical Models in Bioinformatics, The Zoo of Graphical Models in Bioinformatics, Markov Models and DNA Symmetries, Markov Models and Gene Finders, Hybrid Models and Neural Network Parameterization of Graphical Models, The Single-Model Case, Bidirectional Recurrent Neural Networks for Protein Secondary Structure Prediction.

Text Books

1. Pierre Baldi and Søren Brunak, *Bioinformatics: The Machine Learning Approach*, MIT Press.
2. Xiaoming Liu (Ed.), *Machine Learning in Bioinformatics*, Chapman and Hall/CRC.
3. Richard Durbin, Sean R. Eddy, Anders Krogh, Graeme Mitchison, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press.
4. Andreas D. Baxeavanis and B. F. Francis Ouellette (Eds.), *Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins*, Wiley.

Reference Books

1. Pierre Baldi and Søren Brunak, *Bioinformatics: The Machine Learning Approach*, MIT Press.
2. Xiaoming Liu (Ed.), *Machine Learning in Bioinformatics*, Chapman and Hall/CRC.
3. Andreas D. Baxeavanis and B. F. Francis Ouellette (Eds.), *Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins*, Wiley.
4. Mohammed J. Zaki and Wagner Meira Jr., *Data Mining and Analysis: Fundamental Concepts and Algorithms*, Cambridge University Press.
5. Sebastian Raschka, Vahid Mirjalili, *Python Machine Learning*, Packt Publishing.
6. M. Kanehisa, *KEGG: Kyoto Encyclopedia of Genes and Genomes*, Nucleic Acids Research.
7. UniProt Consortium, *UniProt: A Worldwide Hub of Protein Knowledge*, Nucleic Acids Research.
8. Jumper et al., *Highly Accurate Protein Structure Prediction with AlphaFold*, Nature.

9. Zhou & Troyanskaya, *Predicting Effects of Noncoding Variants with Deep Learning-Based Sequence Models*, Nature Methods.
10. R. Durbin, S. Eddy, A. Krogh, G. Mitchison, *Biological Sequence Analysis*, Cambridge University Press.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Foundations of Computational Biology		9 hours
1.1	Biological Data in Digital Symbol Sequences, Genomes—Diversity, Size, and Structure	1
1.2	Proteins and Proteomes	1
1.3	Prediction of Molecular Function and Structure	1
1.4	Machine Learning Foundations: The Probabilistic Framework	2
1.5	Introduction: Bayesian Modeling, The Cox Jaynes Axioms	2
1.6	Bayesian Inference and Induction	1
1.7	Model Structures: Graphical Models and Other Tricks	1
Module 2: Probability and Information Theory		11 hours
2.1	Probabilistic Modeling and Inference	1
2.2	The Simplest Sequence Models, Statistical Mechanics	1
2.3	Machine Learning Algorithms, Dynamic Programming	2
2.4	Gradient Descent, EM/GEM Algorithms	2
2.5	Markov-Chain Monte-Carlo Methods	2
2.6	Simulated Annealing	1
2.7	Evolutionary and Genetic Algorithms, Learning Algorithms: Miscellaneous Aspects	2
Module 3: Neural Networks		10 hours
3.1	Universal Approximation Properties, Priors and Likelihoods	1

3.2	Learning Algorithms: Backpropagation Neural Networks: Applications	2
3.3	Sequence Encoding and Output Interpretation	1
3.4	Sequence Correlations and Neural Networks	2
3.5	Prediction of Protein Secondary Structure	2
3.6	Prediction of Signal Peptides and Their Cleavage Sites	1
3.7	Applications for DNA and RNA Nucleotide Sequences	1
Module 4: Hidden Markov Models		8 hours
4.1	Hidden Markov Models	1
4.2	Prior Information and Initialization	1
4.3	Likelihood and Basic Algorithms	1
4.4	Learning Algorithms	1
4.5	Applications of HMMs: General Aspects	1
4.6	Hidden Markov Models: Applications	1
4.7	Protein Applications DNA and RNA Applications	1
4.8	Advantages and Limitations of HMMs	1
Module 5: Probabilistic Graphical Models in Bioinformatics		7 hours
5.1	The Zoo of Graphical Models in Bioinformatics	1
5.2	Markov Models and DNA Symmetries	2
5.3	Markov Models and Gene Finders	1
5.4	Hybrid Models and Neural Network Parameterization of Graphical Models	1
5.5	The Single-Model Case	1
5.6	Bidirectional Recurrent Neural Networks for Protein Secondary Structure Prediction	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Given a DNA sequence "ATGCATGCATGC", transform it into a feature vector using k-mer (trimer) frequencies and calculate the prior and likelihood for a simple Bayesian classifier that predicts whether this sequence belongs to a coding or non-coding region.

2. Using the Chargaff's rules and given nucleotide frequencies (A=30%, T=30%, C=20%, G=20%), construct a probabilistic model to generate synthetic DNA sequences of length 100 bp. Calculate the posterior probability that a newly observed sequence with these frequencies comes from a eukaryotic versus prokaryotic genome.

Course Outcome 2 (CO 2):

1. Implement the Needleman-Wunsch algorithm (dynamic programming) to align two protein sequences: "HEAGAWGHEE" and "PAWHEAE". Use BLOSUM62 matrix with gap penalty of -8. Show the alignment matrix and traceback.
2. Apply gradient descent to optimize weights in a logistic regression model for predicting disease state from gene expression data of 3 genes (features). Given initial weights $w=[0.1, -0.2, 0.05]$, learning rate $\eta = 0.01$, and 5 training samples, calculate the weight update after one iteration.

Course Outcome 3 (CO 3):

1. Design a feedforward neural network architecture for binary classification of protein sequences as enzymes or non-enzymes. Given sequences of length 50 amino acids encoded as one-hot vectors (20 features per position), specify the input layer size, hidden layer configuration, output layer, and activation functions.
2. For a neural network predicting protein secondary structure (3 classes: helix, sheet, coil) with confusion matrix results: True Helix=45, False Helix as Sheet=5, False Helix as Coil=8; True Sheet=38, False Sheet as Helix=6, False Sheet as Coil=4; True Coil=52, False Coil as Helix=7, False Coil as Sheet=9. Calculate precision, recall, and F1-score for the "Helix" class.

Course Outcome 4 (CO 4):

1. Construct a simple 3-state HMM for gene finding with states: Start, Coding, Non-coding, Stop. Given transition probabilities and emission probabilities for nucleotides, calculate the probability of observing sequence "ATGCCCTAAG" using the Forward algorithm.
2. Apply the Viterbi algorithm to find the most likely state path for DNA sequence "ATGCGAT" in an HMM with 2 states (CpG island, non-CpG), given transition matrix and emission probabilities. Show the Viterbi matrix and traceback.

Course Outcome 5 (CO 5):

1. Design a hybrid model combining a probabilistic graphical model (Markov Random Field) with a neural network for predicting protein-protein interactions. Specify how the graphical model captures domain knowledge and how the neural network learns from sequence features.

2. Implement a bidirectional recurrent neural network (BiRNN) architecture for protein secondary structure prediction. Given an amino acid sequence of length L , describe the forward and backward passes, and how the hidden states from both directions are combined at each position to make the 3-class prediction (H/E/C).



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CSH52

***Course Name: MACHINE LEARNING IN COMPUTATIONAL BIOLOGY
(Honours)***

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define bioinformatics and state its interdisciplinary nature.
2. List any three types of biological data used in bioinformatics.
3. Define a random variable with reference to biological sequences.
4. What is entropy? Why is it important in biological sequence analysis?
5. What is an artificial neural network (ANN)?
6. Define a neuron model used in neural networks.
7. What is a Hidden Markov Model (HMM)?
8. Define match and insert states in a profile HMM.
9. What is a probabilistic graphical model (PGM)?
10. Define a Bayesian Network.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. Explain the role of machine learning in bioinformatics. Discuss how biological data are represented for computational analysis. (14)

OR

12. Describe the bioinformatics problem-solving pipeline. Explain the role of machine learning at different stages. (14)

13. Explain how probabilistic models are used to represent DNA or protein sequences. (14)

OR

14. Given nucleotide probabilities $P(A) = 0.4, P(C) = 0.1, P(G) = 0.1, P(T) = 0.4$ calculate the entropy of the source and interpret its biological significance. (14)

15. (a) Explain the architecture of a feedforward neural network. Describe the function of input, hidden, and output layers. (7)

- (b) Discuss the applications of neural networks in bioinformatics. (7)

OR

16. Explain the working of the backpropagation algorithm in a multilayer neural network. (14)

17. Given the transition and emission probabilities of a simple profile HMM, compute the probability of generating a short biological sequence and explain its relevance in sequence alignment. (14)

OR

18. Consider an HMM with two states and given transition and emission probabilities. Compute the probability of an observation sequence using the forward algorithm and explain its significance. (14)

19. Explain Markov Random Fields and their applications in bioinformatics. (14)

OR

20. Draw and explain a simple Bayesian Network showing dependencies among genes and their expression levels. (14)

B24CSH53	PRINCIPLES OF MODEL CHECKING (Honours)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course covers the basic theory and algorithms for an automatic verification process namely, model checking. Model checking is a formal process for proving the correctness of a hardware/software which can be modelled as a finite-state transition system. This course introduces the topics - finite-state modelling of hardware/software systems, linear-time properties, classification of linear-time properties, Linear Temporal Logic (LTL), a formal language for property specification, LTL model checking algorithm and CTL model checking algorithm. Proving correctness of a hardware/software is essential in safety critical systems in domains such as avionics, health care and automotive.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Analyze and model hardware and software systems using transition systems and apply model checking techniques to verify their correctness and identify system characteristics. (Cognitive Knowledge Level: Understand)
CO 2	Describe finite-state modelling of hardware and software. (Cognitive Knowledge Level: Understand)
CO 3	Identify the linear-time properties required to represent the requirements of a system. (Cognitive Knowledge Level: Apply)
CO 4	Specify a given linear-time property in Linear Temporal Logic (LTL). (Cognitive Knowledge Level: Apply)
CO 5	Perform CTL model checking algorithms to determine whether a given Kripke structure satisfies specific branching-time properties expressed in Computation Tree Logic (CTL). (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	2	1	2	2						
CO 2	3	2	1	1	1						
CO 3	3	2	1	2	2						
CO 4	3	2	1	2	2						
CO 5	3	3	2	3	3						

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (8 hours)

Introduction to Model Checking

System Verification– Hardware and Software Verification, Model Checking, Characteristics of Model Checking. Transition Systems – Transition System, Direct Predecessors and Successors, Terminal State, Deterministic Transition System.

Executions - Execution Fragment, Maximal and Initial Execution Fragment, Execution, Reachable States. Modeling Hardware and Software Systems- Sequential Hardware Circuits, Data Dependent Systems.

MODULE 2 (10 hours)

Linear Time Properties

Linear-Time (LT) Properties - Deadlock. Linear-Time Behavior - Paths and State Graph, Path Fragment, Maximal and Initial Path Fragment, Path. Traces - Trace and Trace Fragment, LT Properties - LT Property, Satisfaction Relation for LT Properties, Trace Equivalence and LT Properties. Safety Properties and Invariants - Invariants, Safety Properties, Trace Equivalence and Safety properties. Liveness Properties - Liveness Property, Safety vs. Liveness Properties. Fairness - Fairness, Unconditional, Weak and Strong Fairness, Fairness Strategies, Fairness and Safety. (Definition and examples only for all topics - no proof required.)

MODULE 3 (8 hours)

Regular Properties

Regular Properties - Model Checking Regular Safety properties - Regular Safety property, Verifying Regular Safety Properties. Automata on Infinite Words - ω -Regular Languages and Properties, Nondeterministic Buchi Automata (NBA), Deterministic Buchi Automata (DBA), Generalised Buchi Automata (Definitions only). Model Checking ω -Regular Properties - Persistence Properties and Product, Nested Depth-First Search (Only algorithms required).

MODULE 4 (9 hours)

Linear Time Logic

Linear Temporal Logic (LTL) - Syntax, Semantics, Equivalence of LTL Formulae, Weak

Until, Release and Positive Normal Form, Fairness, Safety and Liveness in LTL (Definitions only). Automata Based LTL Model Checking (Algorithms and examples only).

MODULE 5 (10 hours)

Introduction to CTL

Computation Tree Logic - Syntax, Semantics, Equivalence of CTL Formulae, Normal Forms for CTL, Expressiveness of CTL vs. LTL, CTL Model Checking-Basic Algorithm, The Until and Existential Always Operator, Time and Space Complexity, Fairness in CTL. **Text**

Books

1. Christel Baier and Joost-Pieter Katoen, Principles of Model Checking, The MIT Press.

Reference Books

1. Logic in Computer Science: Modelling and Reasoning about Systems” by Michael Huth and Mark Ryan, Second Edition.
2. ”Model Checking” by Edmund M. Clarke Jr., Orna Grumberg, and Doron A. Peled, Second Edition.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction to Model Checking		8 hours
1.1	System Verification – Hard- and Software Verification, Model Checking, Characteristics of Model Checking	1
1.2	Transition Systems – Transition System, Direct Predecessors and Successors	1
1.3	Terminal State, Deterministic Transition System	1
1.4	Executions - Execution Fragment, Maximal and Initial Execution Fragment	1
1.5	Execution, Reachable States	1

1.6	Modeling Hardware and Software Systems - Sequential Hardware Circuits	1
1.7	Data Dependent Systems (Lecture 1)	1
1.8	Data Dependent Systems (Lecture 2)	1
1.9	Proximity Measures- Data Objects, Attribute types, Dissimilarity and Similarity measures.	1
Module 2: Linear Time Properties		10 hours
2.1	Linear-Time (LT) Properties - Deadlock	1
2.2	Linear-Time Behavior - Paths and State Graph, Path Fragment	1
2.3	Maximal and Initial Path Fragment, Path	1
2.4	Traces - Trace and Trace Fragment	1
2.5	LT Property, Satisfaction Relation for LT Properties, Trace Equivalence and LT Properties.	1
2.6	Invariants.	1
2.7	Safety Properties, Trace Equivalence and Safety properties	1
2.8	Liveness Property, Safety vs. Liveness Properties	1
2.9	Fairness, Unconditional, Weak and Strong Fairness	1
2.10	Fairness Strategies, Fairness and Safety	1
Module 3: Regular Properties		8 hours
3.1	Regular Properties - Model Checking Regular Safety properties - Regular Safety property	1
3.2	Verifying Regular Safety Properties	1
3.3	Automata on Infinite Words - ω -Regular Languages and Properties	1
3.4	Nondeterministic Buchi Automata (NBA), Deterministic Buchi Automata (DBA), Generalised Buchi Automata	1
3.5	Model Checking ω -Regular Properties - Persistence Properties and Product - Lecture 1	1
3.6	Persistence Properties and Product - Lecture 2	1
3.7	Nested Depth-First Search (Lecture 1)	1
3.8	Nested Depth-First Search (Lecture 2).	1
Module 4: Linear Time Logic		9 hours

4.1	Linear Temporal Logic – Linear Temporal Logic (LTL) - Syntax	1
4.2	Semantics - Lecture 1	1
4.3	Semantics - Lecture 2	1
4.4	Equivalence of LTL Formulae, Weak Until	1
4.5	Release and Positive Normal Form.	1
4.6	Fairness, Safety and Liveness in LTL.	1
4.7	Automata Based LTL Model Checking - Lecture 1	1
4.8	Automata Based LTL Model Checking - Lecture 2	1
4.9	Automata Based LTL Model Checking - Lecture 3.	1
Module 5: Computation Tree Logic		10 hours
5.1	Computation Tree Logic -Syntax	1
5.2	Computation Tree Logic - Semantics	1
5.3	Equivalence of CTL Formulae	1
5.4	Normal Forms for CTL.	1
5.5	Expressiveness of CTL vs. LTL	1
5.6	CTL Model Checking-Basic Algorithm.	1
5.7	The Until and Existential Always Operator.	1
5.8	Time and Space Complexity.	1
5.9	Fairness in CTL (Lecture 1)	1
5.10	Fairness in CTL (Lecture 2).	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Consider a simple digital circuit controlling a single LED light. The circuit has a button input. When the button is pressed, the LED changes its state (On to Off, or Off to On). The circuit initially starts with the LED Off.
 - a) Construct a "Transition System" to model this LED control circuit. Define your States (e.g., LED state, button state), Atomic Propositions (e.g., LED-on, button-pressed), and the possible Transitions. Indicate which state is the initial state.
 - b) From your constructed transition system, determine if it is a "Deterministic Transition System." Justify your answer based on the definition.

- c) Identify all "Terminal States" in your transition system, if any. If there are none, explain why.

Course Outcome 2 (CO 2):

1. Consider a message delivery system. The sender s is trying to send a series of messages to the receiver r in such a way that the $(i+1)^{\text{st}}$ message is sent only after the i^{th} message is delivered. There is a possibility of error in sending a message and in that case, s keeps on trying until it is able to send the message. Express this process as a transition system.

Course Outcome 3 (CO 3):

1. Consider a shared memory segment s protected using a mutex lock variable m . Two processes p_1 and p_2 are trying to access s . Find the Linear Time properties of the system which will ensure safety, liveness and fairness.

Course Outcome 4 (CO 4):

1. Express the Linear Time properties found in the above system using LTL.

Course Outcome 5 (CO 5):

1. Outline the "Basic Algorithm" for CTL model checking. Describe the general iterative approach used to determine the set of states satisfying a formula, particularly for operators like $EF\ p$ or $EG\ p$.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2026

Course Code: B24CSH53

Course Name: PRINCIPLES OF MODEL CHECKING(Honours)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Construct a deterministic transition system for a light switch that toggles ON and OFF.
2. Differentiate between hardware and software verification.
3. Explain the difference between safety and liveness properties.
4. Write an example trace for a system that violates liveness.
5. Define ω -regular language and ω -regular property.
6. Apply the Nested Depth-First Search algorithm steps for detecting an accepting cycle.
7. Define syntax and semantics of Linear Temporal Logic (LTL).
8. Convert a given LTL formula into Positive Normal Form.
9. Differentiate between CTL and LTL in terms of expressiveness.
10. Apply the CTL model checking algorithm to a small transition system to verify a property.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain in detail the various phases of the model checking process. (7)
(b) Apply the concept of transition systems to represent a real-world process such as an elevator control. (7)

OR

12. (a) Design a simple hardware model and derive its transition system and reachable states. (14)
13. (a) Using examples, explain trace equivalence and LT properties. (7)
(b) What is trace equivalence in Transition Systems? Give an example to show that if two transition systems have the same trace equivalence, they satisfy the same LT properties. (7)

OR

14. (a) Give the transition system for the fault tolerant variant of the dining philosophers problem. (4)
(b) Explain linear-time behavior using paths and state graphs with suitable examples. (10)
15. (a) Apply the nested DFS algorithm to verify an ω -regular property.. (7)
(b) Describe generalized Büchi automata and its use in verifying ω -regular properties. (7)

OR

16. (a) What is a persistence property? Give an example. (6)
(b) Compare NBA, DBA, and GBA with examples. (8)
17. (a) Write LTL formulas for the following:
(a) A request is eventually granted (4)
(b) The system never enters an error state (4)
(b) Illustrate the equivalence of LTL formulae using examples. (10)

OR

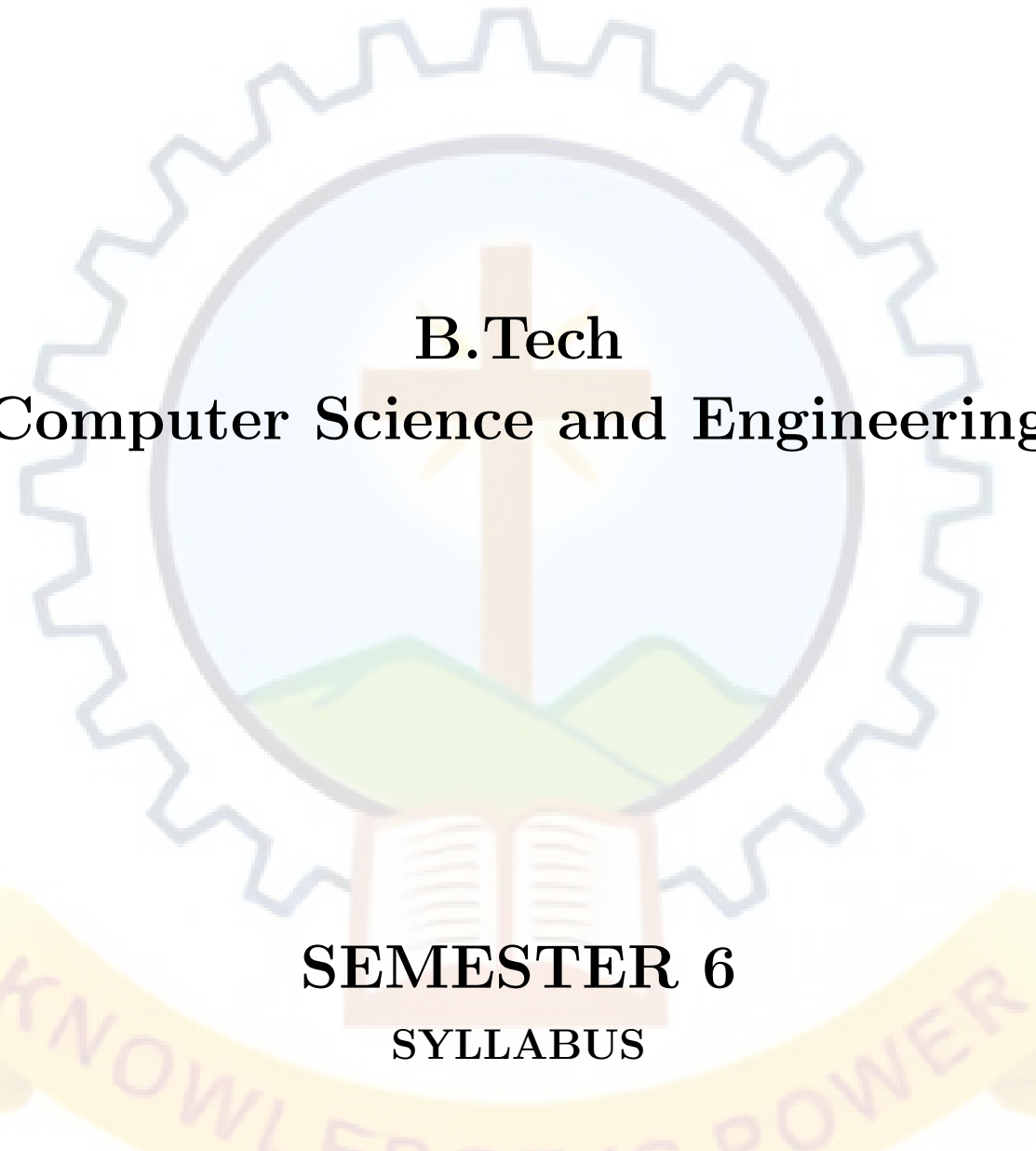
18. (a) Explain the “Until” and “Release” operators in LTL. (7)
(b) Explain automata-based LTL model checking algorithms with suitable examples. (7)
19. (a) What are normal forms for CTL formulas? (7)
(b) Explain the time and space complexity of CTL model checking. (7)

OR

20. Explain in detail the syntax, semantics, and expressiveness of CTL with examples. (14)

MAR ATHANASIOUS COLLEGE OF ENGINEERING

Government Aided, Autonomous Institution
Kothamangalam, Kerala, India

The logo of Mar Athanasius College of Engineering is a circular emblem. It features a large gear-like outer border. Inside the circle, there is a light blue background with a central orange cross. Below the cross are green hills. At the bottom of the circle is an open book. A yellow banner with the text 'KNOWLEDGE IS POWER' in purple capital letters curves around the bottom of the emblem.

B.Tech Computer Science and Engineering

SEMESTER 6 SYLLABUS

SEMESTER 6

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24CS3T06	SECURITY IN COMPUTING	3-1-0-5	4	4
B	B24CS3T07	DATA ANALYTICS SYSTEMS	3-1-0-5	4	4
C	B24CS3T08	CLOUD COMPUTING	3-1-0-5	4	4
D	B24CS3T09	COMPUTER GRAPHICS AND IMAGE PROCESSING	3-1-0-5	4	4
E	B24CS3P2x	PROGRAMME ELECTIVE II	2-1-0-4	3	3
F	B24CS3G1x	OPEN ELECTIVE I	2-1-0-4	3	3
G	B24CS3L09	MICROPROCESSOR LAB	0-0-3-2	3	2
H	B24CS3L10	MINI PROJECT	0-0-3-2	3	2
M	B24CSM6x	MINOR	3-1-0-5	4	4
N	B24CSH6x	HONOURS	3-1-0-5	4	4
TOTAL				36	26

PROGRAMME ELECTIVE II

COURSE NO.	COURSES
B24CS3P21	OBJECT MODELING AND DESIGN (Bucket- I)
B24CS3P22A	CRYPTOGRAPHY AND NETWORK SECURITY (A) (Bucket- II)
B24CS3P23	LAN TECHNOLOGIES (General-I)
B24CS3P24	WEB MINING (General-II)

OPEN ELECTIVE I

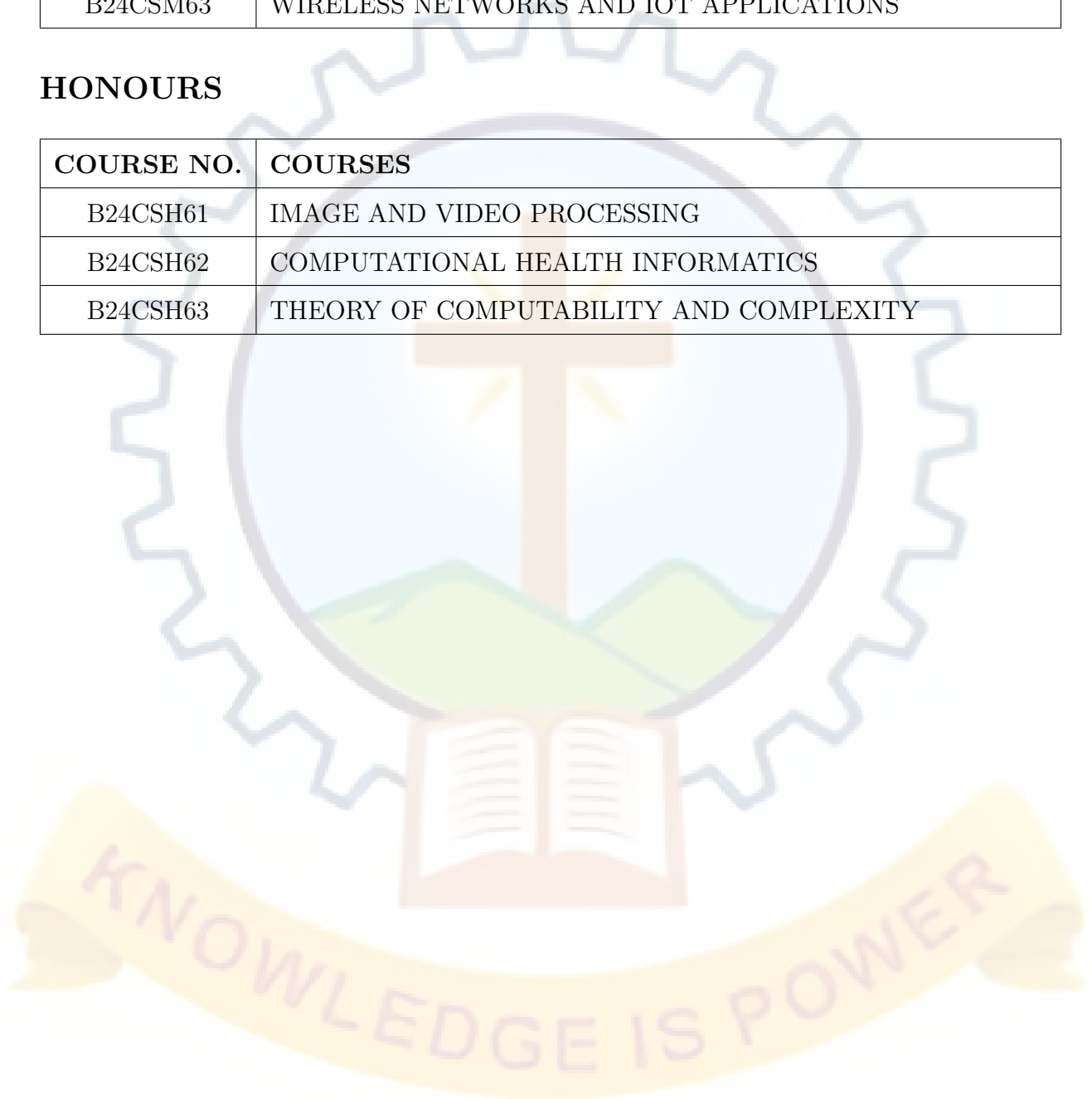
COURSE NO.	COURSES
B24CS3G11	INTRODUCTION TO DATA STRUCTURES
B24CS3G12	COMPUTER GRAPHICS
B24CS3G13	DIGITAL IMAGE PROCESSING

MINOR

COURSE NO.	COURSES
B24CSM61	WEB TECHNOLOGIES
B24CSM62	PYTHON FOR MACHINE LEARNING
B24CSM63	WIRELESS NETWORKS AND IOT APPLICATIONS

HONOURS

COURSE NO.	COURSES
B24CSH61	IMAGE AND VIDEO PROCESSING
B24CSH62	COMPUTATIONAL HEALTH INFORMATICS
B24CSH63	THEORY OF COMPUTABILITY AND COMPLEXITY



B24CS3T06	SECURITY IN COMPUTING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course helps the learners to explore various algorithms to offer confidentiality, integrity, authentication and non-repudiation services and different attacks on system security with their countermeasures. It covers classical encryption techniques, symmetric and public key crypto-systems, key distribution techniques, authentication functions, intruders, malicious software, and DDoS attacks. The concepts covered in this course enable the learners in effective use of cryptographic algorithms and appropriate countermeasures for securing real-life applications.

Prerequisites

A fundamental knowledge in mathematical foundations of security.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Identify the security services provided against different types of security attacks. (Cognitive Knowledge Level: Understand)
CO 2	Describe classical encryption techniques and explain how they support information hiding. (Cognitive Knowledge Level: Understand)
CO 3	Illustrate symmetric/asymmetric key cryptosystems for secure communication. (Cognitive Knowledge Level: Apply)
CO 4	Explain message integrity and authentication methods in a secure communication scenario. (Cognitive Knowledge Level: Understand)
CO 5	Interpret public/secret key distribution techniques for secure communication and identify the effects of intruders, malicious software and distributed denial of service attacks on system security. (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1		1		2	1	1	1		1
CO 2	2	1		1					1		2
CO 3	3	3	3	3	2	3	1	2	3	2	3
CO 4	2	3	3	3	3	3	3	3	3	3	3
CO 5	3	3	3	3	3	3	3	3	3	2	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Basics of Security and Traditional Cryptosystems:

OSI security architecture – Security attacks, Services, Mechanisms. Cryptography vs Cryptanalysis. Classical encryption techniques – Symmetric cipher model. Substitution ciphers – Monoalphabetic vs Polyalphabetic ciphers, Caesar cipher, Vigenere cipher. Transposition ciphers – Double transposition. Steganography – Basic concepts.

MODULE 2 (9 hours)

Modern Symmetric Key Cryptosystems:

Symmetric key ciphers – Block vs Stream ciphers, Feistel ciphers. Data Encryption Standard (DES) – Structure, Key generation, Design criteria, Weaknesses. Advanced Encryption Standard (AES) – Structure, Key expansion. Stream ciphers – Structure, RC4.

MODULE 3 (9 hours)

Public Key Cryptosystems:

Introduction to public key cryptosystems – Principles, Applications, Requirements, Conventional vs Public key cryptosystems. RSA cryptosystem – Algorithm, Security, Attacks. Key Management – Diffie-Hellman key exchange – Algorithm, Man-in-the-middle attack. Elliptic Curve Cryptography (ECC) – Key exchange using ECC.

MODULE 4 (9 hours)

Message Integrity and Authentication:

Hash functions – Security requirements, Secure Hash Algorithm (SHA-512). Message Authentication Code (MAC) – Requirements, Uses, Hash-based MAC (HMAC), Cipher-based MAC (CMAC). Digital signatures – Attacks, Forgeries, Requirements, Direct vs Arbitrated digital signatures, Digital Signature Standard (DSS).

MODULE 5 (9 hours)

Key Distribution and System Security:

Web Security – Considerations, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET) – Basic concepts. System security – Intruders, Intrusion techniques. Malicious software – Viruses, Related threats. Distributed Denial of Service (DDoS) attacks – Types, Countermeasures.

Text Books

1. William Stallings, *Cryptography and Network Security: Principles and Practice*, 4th Edition, Pearson Education.
2. Behrouz A. Forouzan, *Cryptography and Network Security*, 3rd Edition, Tata McGraw-Hill.

Reference Books

1. Charles P. Pfleeger, Shari Lawrence Pfleeger, Jonathan Margulies, *Security in Computing*, 5th Edition, Prentice Hall.
2. G.A. Jones & J.M. Jones, *Elementary Number Theory*, Springer UTM, 2007.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Basics of Security and Traditional Cryptosystems		9 hours
1.1	OSI security architecture – Security attacks	1
1.2	OSI security architecture – Security services	1
1.3	OSI security architecture – Security mechanisms	1
1.4	Cryptography vs Cryptanalysis	1
1.5	Classical encryption techniques – Symmetric cipher model	1
1.6	Substitution ciphers – Monoalphabetic vs Polyalphabetic, Caesar cipher	1
1.7	Substitution ciphers – Vigenere cipher	1
1.8	Transposition ciphers – Double transposition	1
1.9	Steganography – Basic Concepts	1
Module 2: Modern Symmetric Key Cryptosystems		9 hours
2.1	Symmetric key ciphers – Block vs Stream ciphers	1
2.2	Feistel ciphers	1
2.3	Data Encryption Standard (DES) – Structure	1

2.4	DES – Key generation	1
2.5	DES – Design criteria	1
2.6	DES – Weaknesses	1
2.7	Advanced Encryption Standard (AES) – Structure	1
2.8	AES – Key expansion	1
2.9	Stream ciphers – Structure, RC4	1
Module 3: Public Key Cryptosystems		9 hours
3.1	Introduction to public key cryptosystems – Principles, Applications	1
3.2	Public key cryptosystems – Requirements; Conventional vs Public key systems	1
3.3	RSA cryptosystem – Algorithm	1
3.4	RSA cryptosystem – Security aspects	1
3.5	RSA cryptosystem – Attacks	1
3.6	Diffie–Hellman key exchange – Algorithm	1
3.7	Diffie–Hellman – Man-in-the-middle attack	1
3.8	Elliptic Curve Cryptography (ECC) – Basic concepts	1
3.9	ECC – Key exchange using ECC	1
Module 4: Message Integrity and Authentication		9 hours
4.1	Hash functions – Security requirements	1
4.2	Hash functions – Secure Hash Algorithm (SHA-512)	1
4.3	Message Authentication Code (MAC) – Requirements	1
4.4	Message Authentication Code (MAC) – Uses	1
4.5	Hash-based MAC (HMAC), Cipher-based MAC (CMAC)	1
4.6	Digital signatures – Attacks, Forgeries	1
4.7	Digital signatures – Requirements; Direct vs Arbitrated digital signatures	1
4.8	Digital signatures – RSA digital signature, ElGamal digital signature	1
4.9	Digital Signature Standard (DSS)	1
Module 5: Key Distribution and System Security		9 hours
5.1	Web Security – Considerations	1

5.2	Secure Socket Layer (SSL) and Transport Layer Security (TLS)	1
5.3	Secure Electronic Transaction (SET) – Basic Concepts	1
5.4	System security – Intruders	1
5.5	System security – Intrusion techniques	1
5.6	Malicious software – Viruses	1
5.7	Malicious software – Related threats	1
5.8	Distributed Denial of Service (DDoS) attacks – Types	1
5.9	Distributed Denial of Service (DDoS) attacks – Countermeasures	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. During data transmission, an attacker modifies selected packets. What security service would detect this and how?
2. Distinguish between security attacks, security services, and security mechanisms.

Course Outcome 2 (CO 2):

1. Describe double transposition cipher and show how it hides information.
2. Describe the Caesar cipher and compute ciphertext for the plaintext ATTACK with key = 3.

Course Outcome 3 (CO 3):

1. Using $p = 23$, $g = 5$, Alice's private key $a = 4$, and Bob's private key $b = 3$, apply the Diffie-Hellman key exchange protocol and compute the shared secret key.
2. Given the RSA public key [$(n = 391, e = 3)$] and ciphertext [$C = 52$,] decrypt the message using the RSA decryption steps.

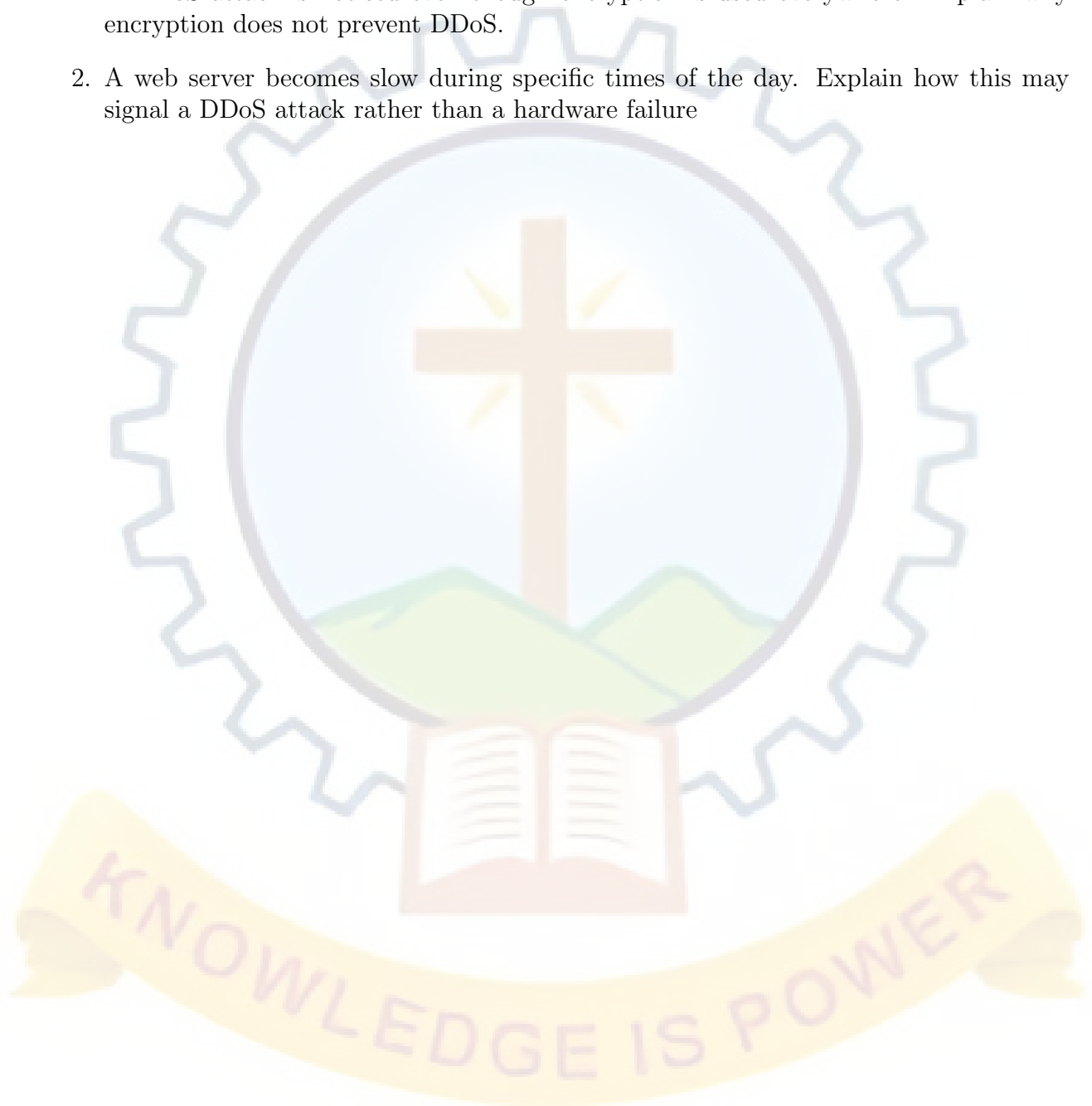
Course Outcome 4 (CO 4):

1. A message is received unchanged, but the sender denies sending it. Identify which mechanism failed and justify.

2. If a system relies only on digital signatures but not MACs, in what scenario could this become inefficient?

Course Outcome 5 (CO 5):

1. A DDoS attack is noticed even though encryption is used everywhere. Explain why encryption does not prevent DDoS.
2. A web server becomes slow during specific times of the day. Explain how this may signal a DDoS attack rather than a hardware failure



MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CS3T06

Course Name: SECURITY IN COMPUTING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Differentiate between passive attack and active attack.
2. Describe the Vigenere cipher. Why is it more secure than the Caesar cipher?
3. Compare stream cipher and block cipher with example.
4. Differentiate between diffusion and confusion.
5. Define the elliptic curve logarithm problem.
6. Given two prime numbers $p = 7$ and $q = 11$, construct the RSA public key (n, e) using $e = 7$. Show all steps.
7. Give the requirements of MAC function.
8. Specify the different types of forgery in digital signature.
9. List three different classes of intruders.
10. What is a computer virus? Explain any two types of related malicious threats.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) You are asked to design a secure message communication system for a small organization. Explain how substitution ciphers and transposition ciphers can be combined to strengthen security, and justify why using both provides better protection against cryptanalysis. (7)
- (b) A plaintext message "MEET AT DAWN" is to be encrypted. Demonstrate how the message is encrypted using
- (a) Caesar cipher with a shift of 4
- (b) Vigenère cipher with keyword "KEY"
- Analyze which method provides better resistance against frequency analysis and why. (7)

OR

12. (a) You are asked to securely hide confidential data inside an image. Explain how Steganography can be applied and what precautions must be taken to avoid detection. (7)
- (b) Differentiate between Cryptography and Cryptanalysis with suitable examples. (7)
13. (a) Explain the complete DES algorithm. Describe its structure, rounds, functions, and the working of the F-function with appropriate diagrams. (7)
- (b) Illustrate the general framework of symmetric key encryption. Explain the roles of encryption algorithm, key, plaintext, ciphertext, and security assumptions with a neat diagram. (7)

OR

14. (a) Using RSA, let $p = 17$, $q = 19$, and $e = 5$: (i) Construct the public and private keys. (ii) Encrypt the message $M = 12$. (iii) Demonstrate the decryption process. (7)
- (b) Explain the fundamental principles and requirements of a public key cryptosystem. Discuss how these relate to confidentiality, authentication, and non-repudiation. (7)
15. (a) Given elliptic curve $E : y^2 = x^3 + 2x + 2 \pmod{17}$, and points $P = (5, 1)$ and $Q = (6, 3)$: (i) Compute $P + Q$. (ii) Compute $2P$. (7)
- (b) Describe the RSA algorithm in detail. Explain the processes of key generation, encryption, and decryption with proper steps. (7)

OR

16. (a) Describe the SHA-512 algorithm in detail. Explain the message preprocessing, padding, message schedule, compression function, and output generation steps. (7)

- (b) Explain the concept of Message Authentication Code (MAC). Discuss the requirements and uses of a MAC in secure communication. (7)
17. (a) Explain the working and structure of the Digital Signature Standard (DSS). Describe the roles of SHA and DSA in the standard. (7)
- (b) Digital signatures ensure non-repudiation. Explain why message authentication codes cannot provide non-repudiation. Support your reasoning with an example scenario. (7)

OR

18. (a) Describe the major components of Secure Electronic Transaction (SET). Explain how SET ensures confidentiality, integrity, and authentication in online payments. (7)
- (b) Explain the different types of intruders in system security. Describe masqueraders, misfeasors, and clandestine users with appropriate examples. (7)
19. (a) Compare viruses, worms, and Trojan horses in terms of structure, propagation method, and damage caused. Give examples for each. (7)
- (b) A malicious program appears to be useful software but secretly sends user data to an attacker. Identify the type of malware and explain how it works, how it spreads, and how it can be prevented. (7)

OR

20. (a) Compare viruses, worms, and Trojan horses in terms of structure, propagation method, and damage caused. Give examples for each. (7)
- (b) A malicious program appears to be useful software but secretly sends user data to an attacker. Identify the type of malware and explain how it works, how it spreads, and how it can be prevented. (7)

B24CS3T07	DATA ANALYTICS SYSTEMS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		2024

Preamble

This course helps the learner to understand the basic concepts of data analytics, mathematics for data analytics, predictive and descriptive analytics of data, Neural Networks and information retrieval techniques for managing data . It enable the learners to perform data analysis on a real world scenario using appropriate tools.

Prerequisites

Data Structures, C/Python

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the key concepts of data analytics. (Cognitive Knowledge Level: Understand)
CO 2	Apply appropriate techniques to convert raw data into suitable format for practical data analytics tasks (Cognitive Knowledge Level: Apply)
CO 3	Apply the concept of Association rule mining in real world scenario (Cognitive Knowledge Level: Apply)
CO 4	Illustrate the basic concepts of neural networks and its practical issues (Cognitive Knowledge Level : Apply)
CO 5	Understand the basics of text representation and text retrieval (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2					1			1		2
CO 2	3	2	1	3	3	1			1		2
CO 3	3	3	2	3	3	2	2	1	2	2	3
CO 4	3	3	2	3	3	2	2	1	3	2	2
CO 5	3	2	1	3	3	2	2	1	2		3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contains 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Introduction to Data Analysis

Introduction to Data Analytics, Analytics Process Model, Analytical Model Requirements, Data Analytics Life Cycle overview.

Descriptive statistics - Measures of central tendency and dispersion, Association of two variables - Discrete variables, Ordinal and Continuous variable, Inductive statistics - Point estimation, Interval estimation, Hypothesis Testing - Basic definitions, t test.

Proximity Measures - Data Objects, Attribute types, Dissimilarity and Similarity measures.

MODULE 2 (7 hours)

Statistical Description of data

Association of Two Variables: Summarizing the Distribution of Two Discrete Variables, Contingency Tables for Discrete Data, Joint, Marginal, and Conditional Frequency Distributions, Measures of Association for Two Discrete Variables, Association Between Ordinal and Continuous Variables.

Mean, Median, Range, Quartiles, Variance, Standard Deviation, and Interquartile Range.

Data Preprocessing - Cleaning, Integration, Reduction.

MODULE 3 (10 hours)

Introduction to Frequent Pattern Mining

Mining Frequent Patterns - Discretization, Transformation, Associations, Correlations, and Apriori Algorithm.

Classification - General approach to classification, Attribute selection measures, Naive Bayesian Classification.

Clustering - K-Means, Agglomerative versus Divisive.

MODULE 4 (9 hours)

Introduction to Neural Networks

Biological Neuron, Perceptron, Multilayer Neural Networks. Activation functions – Sign, Sigmoid, Tanh, ReLU, leaky ReLU, Hard Tanh, Softmax. Loss function. Training a Neural Network with Backpropagation. Practical issues in neural network training. Overfitting, Underfitting, Hyper parameters and Validation sets.

MODULE 5 (10 hours)

Introduction to Information Retrieval

Text Processing: Data Acquisition, Text Extraction and Clean-up, Pre-processing, Feature Engineering, Modeling, Evaluation and Post-Modeling.

Text Representation-Vector Space Models, Basic Vectorization Approaches-One-Hot Encoding, Bag of Words, Bag of N-Grams, TF-IDF.

Distributed Representations- Word Embeddings, Doc2Vec.

Boolean retrieval, Example IR problem, inverted index, processing Boolean queries, tokenization, stemming, phrase queries, vector space model

Text Books

1. Introduction to Statistics and Data Analysis, Christian Heumann and Michael Schomake, Springer 1/e, 2016
2. Jiawei Han and Micheline Kamber, Data Mining Concepts and Techniques Elsevier 3/e, 2012

Reference Books

1. Introduction to Information Retrieval Christopher D. Manning, Raghavan, P., Schutze, H. Cambridge University Press 1/e, 2008
2. Mining Text Data Charu C. Aggarwal, Cheng Xiang Zhai Springer 1/e, 2012
3. Analytics in a Big Data World: The Essential Guide to Data Science and its Business Intelligence and Analytic Trends Bart Baesens John Wiley 1/e, 2013

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction to Data Analysis		9 hours
1.1	Introduction to Data Analytics, Process Model	1
1.2	Analytical Model Requirements, Data Analytics Life Cycle	1
1.3	Descriptive Statistics - Measures of Central Tendency	1

1.4	Descriptive Statistics - Measures of Dispersion	1
1.5	Association of Two Variables - Discrete and Ordinal	1
1.6	Association of Two Variables - Continuous	1
1.7	Inductive Statistics - Point estimation, Interval estimation	1
1.8	Hypothesis Testing - Basic definitions, t test	1
1.9	Proximity Measures-Data Objects, Attribute types, Dissimilarity and Similarity measures. .	1
Module 2: Statistical Description of data		7 hours
2.1	Association of Two Variables:- Summarizing the Distribution of Two Discrete Variables.	1
2.2	Contingency Tables for Discrete Data, Joint, Marginal	1
2.3	Conditional Frequency Distributions	1
2.4	Measures of Association for Two Discrete Variables	1
2.5	Association Between Ordinal and Continuous Variables.	1
2.6	Mean, Median, Range, Quartiles, Variance, Standard Deviation, and Interquartile Range.	1
2.7	Data Preprocessing - Cleaning, Integration, Reduction	1
Module 3: Introduction to Frequent Pattern Mining		10 hours
3.1	Mining Frequent Patterns - Discretization	1
3.2	Mining Frequent Patterns - Transformation	1
3.3	Mining Frequent Patterns - Associations	1
3.4	Mining Frequent Patterns - Correlations	1
3.5	Mining Frequent Patterns - Apriori Algorithm.	1
3.6	Classification - General approach to classification	1
3.7	Attribute selection measures	1
3.8	Naive Bayesian Classification.	1
3.9	Clustering - K-Means	1
3.10	Agglomerative versus Divisive.	1
Module 4: Introduction to Neural Networks		9 hours
4.1	Biological Neuron , Perceptron.	1
4.2	Multilayer Neural Networks	1
4.3	Activation functions – Sign,Sigmoid	1

4.4	Activation functions – Tanh, ReLU, leaky ReLU	1
4.5	Activation functions – Hard Tanh, Softmax, Loss function.	1
4.6	Training a Neural Network with Backpropagation.	1
4.7	Practical issues in neural network training	1
4.8	Overfitting, Underfitting	1
4.9	Hyper parameters and Validation sets.	1
Module 5: Introduction to Information Retrieval		10 hours
5.1	Text Processing :-Data Acquisition	1
5.2	Text Extraction and Clean-up	1
5.3	Pre-processing, Feature Engineering, Modelings	1
5.4	Evaluation and Post-Modeling.	1
5.5	Text Representation-Vector Space Models	1
5.6	,Basic Vectorization Approaches-One-Hot Encoding.	1
5.7	Bag of Words, Bag of N-Grams, TF-IDF.	1
5.8	Distributed Representations- Word Embeddings, Doc2Vec.	1
5.9	Boolean retrieval, Example IR problem, inverted index	1
5.10	processing Boolean queries, tokenization, stemming, phrase queries, vector space mode.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Suppose a company wants to predict customer churn using analytics. Which type of model (descriptive, predictive, prescriptive) would you choose and why?
2. The monthly sales (in Rs.1000s) of a store are:
15 , 18 , 20 , 22 , 25 15,18,20,22,25
Compute the coefficient of variation and explain what it means.

Course Outcome 2 (CO 2):

1. Explain the concept of a contingency table for two discrete variables. How do you construct joint, marginal, and conditional frequency distributions from it? Provide an example with a 2x2 table.

2. Define measures of association for two discrete variables. Compare Chi-square test and Cramer's V with suitable formulas.

Course Outcome 3 (CO 3):

1. Consider a dataset with an attribute 'Age' ranging from 10 to 80. Apply equal-width discretization to divide 'Age' into 4 bins. For a transaction with Age = 35, which bin would it fall into? Explain why discretization is beneficial here.
2. You have the following 2D data points: A=(1,1), B=(2,1), C=(4,3), D=(5,4). Perform one iteration of K-Means clustering with K=2. Assume initial centroids are M1=(1.5,1) and M2=(4.5,3.5). Show the cluster assignments and the new centroids after this iteration. Use Euclidean distance.

Course Outcome 4 (CO 4):

1. A Perceptron has weights $w_1=0.5$, $w_2=0.3$ and a bias $b=-0.6$. Given an input vector $X = [x_1=1, x_2=2]$, calculate the weighted sum (net input). If the activation function is a step function (output 1 if net input ≥ 0 , else 0), what is the output of the Perceptron?
2. Consider a neuron that receives an input 'x' and produces an output 'y'. a) Calculate the output 'y' if $x = -2.0$ and the activation function is ReLU. b) Calculate the output 'y' if $x = -2.0$ and the activation function is Leaky ReLU with $\alpha = 0.01$. c) Explain why Leaky ReLU might be preferred over standard ReLU in some scenarios.

Course Outcome 5 (CO 5):

1. You are building an IR system for a medical research institute. They frequently search for phrases like "heart attack" or "flu symptoms". a) Explain why a standard Bag of Words model would be inadequate for these types of queries. b) Describe how the Vector Space Model in conjunction with TF-IDF weighting could be used to retrieve and rank documents for a query like "heart attack symptoms", ensuring relevance to the phrase.
2. You have a continuous attribute 'Age' with values: 18, 25, 30, 32, 45, 50, 60, 65. a) Apply equal-frequency (equal-depth) discretization to divide 'Age' into 4 bins. b) Discuss a scenario where equal-frequency discretization might be preferred over equal-width.

MODEL QUESTION PAPER

QP CODE:

Pages: 4

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CS3T07

Course Name: DATA ANALYTICS SYSTEMS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. A random sample of 25 students has an average height of 165 cm and a standard deviation of 10 cm. Construct a 95 percent confidence interval for the population mean height.
2. Define the null and alternative hypotheses with an example.
3. Discuss the steps for data integration from multiple sources.
4. Describe three major data reduction methods.
5. Compare the attribute selection measures with the help of an example.
6. Given the following 2D points for clustering: A(1,1), B(2,2), C(3,3), D(8,8), E(9,9), F(10,7), G(1,3), H(2,4). Apply K-Means algorithm to cluster the points into 2 clusters. Initialize the centroids at A(1,1) and D(8,8).
7. Briefly describe the primary function of the axon and dendrites in a biological neuron.
8. Define Underfitting in the context of neural network training.
9. What is a common pitfall when interpreting correlations without considering associations, especially in transactional data?
10. What are the three core components that define a Naive Bayesian Classifier?

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) A hospital wants to use analytics to predict patient readmission within 30 days. Explain how the Analytics Process Model would be applied to solve this problem. Specify data sources, feature engineering, modeling approach, and evaluation metrics. (7)
- (b) A dataset of customer ratings is:
2,3,4,5,5,4,3,2,5, 4,2,3,4,5,5,4,3,2,5,4
- Compute variance and coefficient of variation.
 - Explain what the coefficient of variation indicates about customer consistency.
 - Suggest how this information could be useful to a business.
- (7)

OR

12. (a) Given two customers with features (in normalized scale): Customer A = [0.6, 0.8, 0.7] Customer B = [0.3, 0.9, 0.5]
- Compute the Euclidean, Manhattan, and Cosine similarity.
 - Identify which customers are more similar and justify your answer.
- (7)
- (b) The following data shows the number of advertisements shown (X) and number of product sales (Y): (7)

X (Ads)	2	4	6	8	10
Y (Sales)	10	18	28	40	55

- Compute the Pearson correlation coefficient (r) between X and Y.
 - Interpret the strength and direction of the relationship.
 - Explain how this insight could guide marketing decisions
13. (a) For the dataset: [12, 15, 18, 20, 22, 25, 28, 30, 35, 50] (house prices in 1000s), compute and interpret Mean, Median, Quartiles, Variance, SD, Range, and IQR. Explain why median and IQR are better for this skewed data. (8)
- (b) Five students were ranked by two teachers based on performance: (6)

Student	Teacher A	Teacher B
A	1	2
B	2	1
C	3	4
D	4	3
E	5	5

- i. Compute the Spearman Rank Correlation Coefficient.
- ii. Interpret whether the teachers' rankings are consistent.
- iii. Discuss one real-life situation where ordinal correlation is useful (e.g., satisfaction surveys, ranks).

OR

14. (a) The following are the marks of students in a test:

45,60,75,80,55,65,70,85,90,50,45,60,75,80,55,65,70,85,90,50

- i. Compute Mean, Median, Range, Variance, Standard Deviation and Interquartile Range (IQR)
- ii. Interpret the variability in students' performance.
- iii. Comment on whether the data is consistent or highly dispersed.

(8)

- (b) A supermarket chain collects transaction data from multiple branches. Each branch uses different formats for storing date, product name, and customer ID. Some entries have missing sales amounts or incorrect units. (6)

Tasks:

1. Identify possible data quality issues (noise, redundancy, inconsistency).
2. Show step-by-step how to perform:
 - i. Data cleaning (handling missing and incorrect values)
 - ii. Data integration (merging data from all branches into one consistent dataset)
 - iii. Data reduction (applying aggregation to get monthly sales instead of daily).

15. (a) Consider a retail transaction dataset with the following sample transactions (Transaction ID: Items):

T1: Milk, Bread, Butter, Eggs
T2: Milk, Bread, Diapers, Beer
T3: Milk, Bread, Butter
T4: Bread, Diapers, Beer
T5: Milk, Diapers, Beer, Butter

Mine frequent patterns and associations.

(7)

- (b) The Price values for the items: Milk: 3.5, Bread: 2.0, Butter: 4.0, Eggs: 1.5, Diapers: 5.0, Beer: 2.5. Perform discretization on the Price attribute by binning into three equal-frequency bins: Low (< 2.5), Medium (2.5-4.0), High (> 4.0). Transform the dataset to include these discretized price levels for each item in transaction. (7)

- i. Insert a new employee record into the table.
- ii. Update the salary of an employee based on their EmpID.
- iii. Delete an employee record who has left the company.
- iv. Retrieve the names and salaries of employees working in the "IT" department.

OR

16. (a) Apply Naive Bayesian Classification to predict the class for a test instance: Age=young, income=medium, student=yes, credit rate=excellent. (8)
- (b) Explain the general approach to classification and find the Information Gain of Age attribute for a decision tree. (6)
17. (a) Imagine you are training a neural network to predict house prices. After initial training, you observe that both your training error and validation error are consistently high.
- What practical issue is this indicative of?
 - Propose two different strategies (mentioning relevant hyperparameters or architectural changes) to address this issue. (7)
- (b) You are tasked with designing a neural network to classify images of cats and dogs.
- Suggest a suitable architecture (number of input, hidden, and output neurons) if the images are 64x64 pixels in grayscale. Justify your choice for the input and output layers.
 - Explain why a single-layer Perceptron would likely fail for this task. (7)

OR

18. (a) A Perceptron has weights $w_1=0.5$, $w_2=0.3$ and a bias $b=-0.6$. Given an input vector $X = [x_1=1, x_2=2]$, calculate the weighted sum (net input). If the activation function is a step function (output 1 if net input ≥ 0 , else 0), what is the output of the Perceptron? (7)
- (b) You are fine-tuning a pre-trained neural network. Explain how the use of a Validation Set helps you decide on the optimal Learning Rate and Number of Epochs for your training process. (7)
19. (a) You are analyzing customer shopping behavior to segment them.
- Explain a scenario where Agglomerative Hierarchical Clustering would be a more suitable choice than K-Means.
 - What visual tool is typically used to interpret the results of hierarchical clustering, and what information does it provide? (7)
- (b) Explain the complete workflow of extracting meaningful insights from raw textual data, from Data Acquisition to Post-Modeling. Elaborate on the significance of Text Extraction and Clean-up in this pipeline. (7)

OR

20. (a) Explain the core principles of Naive Bayesian Classification, detailing the underlying "naïve" assumption and how it impacts the algorithm's calculation and interpretation. Discuss a real-world scenario where its simplicity is a significant advantage. (7)
- (b) Explain the concept of Proximity Measures in data analysis. Describe at least three different proximity measures, outlining the type of data they are best suited for and their underlying mathematical principle. (7)

B24CS3T08	CLOUD COMPUTING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		2024

Preamble

The aim of this course is to provide students with exposure to the cloud computing domain. Students will gain a basic understanding of cloud computing architecture, cloud mechanisms, and techniques used for cloud deployment. The students are expected to be able to deploy virtual machines, understand and work with containerization tools, and gain insight into the modern, heavily cloud-reliant, IT industry. After completion of the course, students shall be able to integrate the modern cloud into their work-flow.

Prerequisite

NIL

Course Outcomes

After the completion of the course, the student will be able to:

CO1	Explain the evolution of cloud computing and differentiate cloud service and deployment models, and analyze cloud-native architectural principles. (Cognitive Knowledge Level: Understand)
CO2	Describe virtualization and containerization concepts, compare virtual machines and containers from an architectural perspective, and interpret container orchestration and configuration patterns (Cognitive Knowledge Level: Analyze)
CO3	Illustrate DevOps principles and practices, explain CI/CD, Infrastructure-as-Code, and configuration management concepts, and evaluate modern cloud resource management (Cognitive Knowledge Level: Analyze)
CO4	Classify cloud platform services across compute, storage, networking, and database domains, and explain serverless and event-driven architectures along with the role of cloud APIs and SDKs in service interaction. (Cognitive Knowledge Level: Analyze)
CO5	Explain core cloud security principles including shared responsibility, IAM, threat concepts, least-privilege access, API security, encryption, and key management, and be able to understand working with cloud service models (IaaS, PaaS, SaaS) from a cloud consumer perspective. (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	1		1	1			2	2	2
CO 2	2	3	2	2	1				1		1
CO 3	2	3	3	3	2	1		2	1	2	1
CO 4	2	3	3	3	2	1		1	2	2	2
CO 5	2	3	3	3	2	3	3		1		1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contains 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE-1 (9 hours)

Foundations of Cloud and Cloud-Native Computing :

Evolution and Enabling Technologies of Cloud computing, Basic Concepts and Terminology, NIST reference model, Cloud characteristics, Cloud delivery models IaaS PaaS SaaS, Cloud deployment models public, private, hybrid, multi-cloud, Cloud-native principles : Stateless design and Service decomposition, Microservices v/s Monolithic architectures, Overview of major cloud platform offerings(AWS, Azure, Google Clouds).

MODULE-2 (9 hours)

Virtualization, Containers and Orchestration :

Virtualization theory, CPU-Memory and I/O virtualization, Hypervisor or Virtual Machine Monitor, Machine or Server Level Virtualization, OS-level virtualization, Emulation, Other types of Virtualizations: Network, Storage and Desktop Virtualizations. Understanding Containerization, Virtualization and Containerization, Understanding Containers, Understanding Container Images, Multi-Container Types, Container architecture and image layering, Comparison of containers and virtual machines, Orchestration patterns explained through Kubernetes architecture (Nodes, pods, deployments and services)

MODULE-3 (8 hours)

Resource Management and DevOps :

Resource Management - Resource Pooling, Sharing, Provisioning; Scaling in Cloud and Scaling Strategies; Auto Scaling in Cloud, Horizontal and Vertical Scaling, Capacity Planning, Load Balancing; DevOps as a set of principles and practices unifying development and operations, Continuous Integration and Continuous Delivery, Automation measures and sharing practices within DevOps culture, Infrastructure as Code with reference model examples, Configuration management concepts, Deployment strategies including blue-green and canary, GitOps introduced as model of declarative operations.

MODULE-4 (10 hours)

Cloud Platform Services and Serverless Computing :

Cloud compute services architecture: Virtual machine and Container orchestration services, Compute services offered by major cloud providers(AWS, Azure, Google Clouds); Cloud storage services: Object, Block and File storage concepts, Popular cloud storage services; Cloud network services: Virtual networks Subnets, Gateways, Serverless computing concepts(Function-as-a-service and Event-driven workflows), Relational NoSQL and Distributed database models, Conceptual overview of cloud SDKs and APIs used for cloud service interaction.

MODULE-5 (9 hours)

Cloud Security and Working with Clouds :

Fundamentals of cloud security architecture including shared responsibility, Identity and Access Management; Basic Security Terminology: Basic Threat Terminology, Threat Agents, Common Threats; Least privilege API security principles, Encryption fundamentals and Key management practices; Working with clouds: The Cloud Consumer Perspective of delivery models(IaaS, PaaS and SaaS) Cost Metrics and Pricing Models: Business cost, Cloud usage costs, Cost management considerations.

Text Books

1. Thomas Erl, Cloud Computing: Concepts, Technology & Architecture, 2nd Edition
2. Rajkumar Buyya et al., Mastering Cloud Computing
3. Nigel Poulton, Docker Deep Dive

Reference Books

1. Gene Kim et al., The DevOps Handbook
2. Brendan Burns et al., Designing Distributed Systems
3. Official cloud provider fundamentals documentation from AWS Azure GCP

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module -1: Foundations of Cloud and Cloud-Native Computing		9 hours
1.1	Evolution of cloud computing	1
1.2	NIST reference model, cloud characteristics	1
1.3	cloud delivery models: IaaS PaaS SaaS,	1
1.4	Cloud deployment models: Public Private Hybrid,	1
1.5	Cloud-native principles including stateless design and service decomposition	1
1.6	Architectural patterns including API gateways and service meshes	2
1.7	Overview of major cloud platform offerings.	2
Module -2: Virtualization, Containers and Orchestration		9 hours
2.1	Virtualization theory including CPU, Memory and I/O virtualization	1
2.2	Server virtualization and hypervisor models	2
2.3	Monolithic vs Microservice architectures	1
2.4	Introduction to container architecture and image layering	1
2.5	Comparison of containers and virtual machines	1
2.6	Orchestration patterns explained through Kubernetes architecture	2
2.7	G concept of configuration objects such as ConfigMaps Secrets Ingress, MicroVMs and WASM.	1
Module -3: DevOps, Automation and Infrastructure-as-Code		8 hours
3.1	DevOps as a set of principles and practices	2
3.2	Continuous Integration(CI) and Continuous Delivery(CD)	1
3.3	Automation measures and sharing practice	1
3.4	Infrastructure as Code with reference model examples	2
3.5	GitOps introduced as model of declarative operations.	1

3.6	Deployment strategies including blue-green and canary	1
Module -4: Cloud Platform Services and Serverless Computing		10 hours
4.1	Cloud compute services theory	1
4.2	Cloud storage theory: object block and file storage concepts.	2
4.3	Load balancer : Theory	1
4.4	Cloud networking concepts: Virtual network subnets, gateways	1
4.5	Serverless computing as architectural model	2
4.6	Familiarization with the services of major Cloud Service Providers(AWS, Azure, Google Cloud)	1
4.7	Relational NoSQL and distributed database models,	1
4.8	Conceptual overview of cloud SDKs and APIs.	1
Module -5: Cloud Security, Observability and Reliability		9 hours
5.1	Concepts of monitoring logging and tracing	1
5.2	Distributed tracing theory	1
5.3	Reliability engineering foundations including SLIs, SLOs and error budgets	1
5.4	Redundancy and health check concepts	1
5.5	Fundamentals of cloud security architecture including shared responsibility, Identity and Access Management	2
5.6	Least privilege API security principles, Encryption fundamentals and key management practices	2
5.7	Risk and compliance framework	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the evolution of cloud computing from traditional data centers to modern cloud platforms.
2. Differentiate IaaS, PaaS, and SaaS using suitable examples and use cases.

Course Outcome 2 (CO 2):

1. Compare virtual machines and containers with respect to architecture, resource utilization, and deployment.
2. Interpret the functions of container orchestration platforms in managing distributed applications.

Course Outcome 3 (CO 3):

1. Illustrate the DevOps lifecycle and the relationship between development and operations teams.
2. Evaluate different approaches to cloud resource provisioning, monitoring, and optimization.

Course Outcome 4 (CO 4):

1. Explain the operation and benefits of serverless and event-driven architectures.
2. Analyze the role of cloud APIs and SDKs in integrating and managing cloud services.

Course Outcome 5 (CO 5):

1. Analyze the significance of IAM, API security, encryption, and key management in protecting cloud resources.
2. Differentiate the security responsibilities of cloud consumers in IaaS, PaaS, and SaaS environments.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CS3T08

Course Name: CLOUD COMPUTING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Explain any three essential characteristics of cloud computing according to the NIST model.
2. Differentiate between Microservices architecture and Monolithic architecture.
3. What is a Hypervisor? Explain its role in virtualization.
4. Compare containers and virtual machines.
5. Distinguish between Horizontal Scaling and Vertical Scaling
6. What is Continuous Integration (CI)? Mention its benefits.
7. Differentiate between Object Storage and Block Storage.
8. What is Serverless Computing? Give an example.
9. Explain the principle of Least Privilege in cloud security.
10. Differentiate between IaaS, PaaS and SaaS from a cloud consumer's perspective.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the evolution of cloud computing and discuss the enabling technologies that led to its emergence. (8)
(b) Describe the cloud delivery models IaaS, PaaS and SaaS with suitable examples. (6)

OR

12. (a) Explain the NIST reference model and the essential characteristics of cloud computing. (7)
(b) Discuss cloud deployment models: Public, Private, Hybrid and Multi-cloud. (7)
13. (a) Explain CPU, Memory and I/O virtualization with suitable illustrations. (8)
(b) Discuss the architecture and functions of a Hypervisor (Virtual Machine Monitor). (6)

OR

14. (a) Explain containerization and discuss the architecture of containers with image layering. (8)
(b) Describe Kubernetes architecture with reference to Nodes, Pods, Deployments and Services. (6)
15. (a) Explain resource pooling, resource provisioning and resource sharing in cloud environments. (7)
(b) Discuss load balancing and auto-scaling mechanisms used in cloud computing. (7)

OR

16. (a) Explain DevOps principles and discuss Continuous Integration and Continuous Delivery practices. (8)
(b) Describe Infrastructure as Code (IaC), configuration management and GitOps concepts. (6)
17. (a) Explain cloud compute services architecture and discuss virtual machine and container orchestration services. (8)
(b) Compare the compute services offered by AWS, Microsoft Azure and Google Cloud Platform. (6)

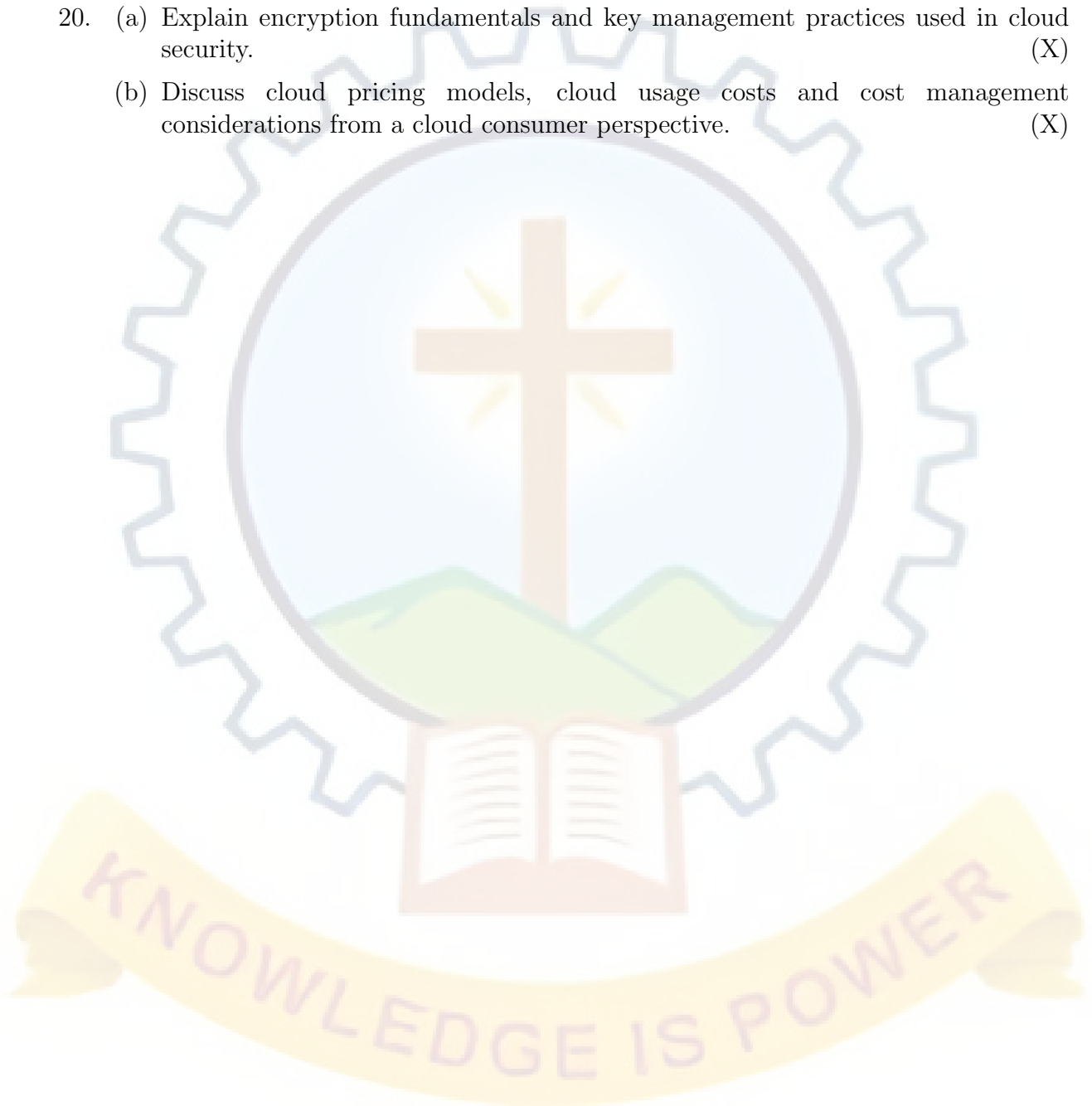
OR

18. (a) Explain cloud storage services and discuss Object, Block and File storage concepts with examples. (8)
(b) Discuss serverless computing, Function-as-a-Service (FaaS) and event-driven workflows. (6)

19. (a) Explain the shared responsibility model and Identity and Access Management (IAM) in cloud security. (8)
- (b) Discuss common cloud threat agents and threats encountered in cloud environments. (6)

OR

20. (a) Explain encryption fundamentals and key management practices used in cloud security. (X)
- (b) Discuss cloud pricing models, cloud usage costs and cost management considerations from a cloud consumer perspective. (X)



B24CS3T09	COMPUTER GRAPHICS AND IMAGE PROCESSING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		

Preamble

This course is designed to impart fundamental knowledge of Computer Graphics and Image Processing essential for the creation, manipulation, and analysis of visual information. The syllabus covers core topics such as graphics systems and geometric transformations, digital image representation, sampling and quantization, spatial domain operations, image enhancement, segmentation, and edge detection. The knowledge gained from this course will enable students to pursue advanced studies and professional opportunities in areas including computer vision, multimedia systems, visualization, and intelligent imaging technologies.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain fundamentals of computer graphics systems, transformations, and rendering pipeline.(Cognitive Knowledge Level: Understand)
CO 2	Apply viewing, modeling, and rendering techniques to design 2D and 3D graphical systems considering practical constraints.(Cognitive Knowledge Level: Apply)
CO 3	Apply digital image representation, sampling, quantization, and pixel relationships . (Cognitive Knowledge Level: Apply)
CO 4	Apply spatial domain image enhancement and filtering techniques.(Cognitive Knowledge Level: Apply)
CO 5	Apply image segmentation and edge detection techniques for advanced image analysis and interpretation using critical, research-oriented approaches ,real-world applications and ethical consideration.(Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2		1					1		
CO 2	2	3	2	2	1				1		
CO 3	3	3	3	3	1				1		2
CO 4	3	3	3	3	3				1		2
CO 5	3	3	3	3	3				1		2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	30	30	30
Apply	50	50	50
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (10 hours)

Computer Graphics - Foundations, Mathematics & 2D Graphics

Overview of Computer Graphics Systems, Applications, components and Graphics pipeline. Coordinate systems, vectors, matrices, homogeneous coordinates, transformation composition, Raster graphics algorithms: Bresenham line drawing and midpoint circle algorithm. 2D geometric transformations: translation, scaling, rotation, Windowing, viewport mapping, and 2D viewing pipeline.

MODULE 2 (10 hours)

Computer Graphics – 3D Graphics, Viewing and Rendering

3D transformations and matrix representation, Camera model and viewing pipeline, Orthographic and perspective projections, Polygonal modeling, vertex attributes, normals, and back-face culling, Hidden surface removal: Painter's algorithm and Z-buffer (conceptual), Basic illumination and shading models (Flat, Gouraud, Phong – overview)

MODULE 3 (9 hours)

Image Processing – Image Fundamentals and Representation

Introduction to image processing and applications, Image as 2D data, Binary, grayscale, and color image representation, Components of an image processing system, Fundamental steps in digital image processing, Coordinate conventions, Sampling and quantization, Spatial and gray-level resolution

MODULE 4 (8 hours)

Image Processing – Spatial Domain Operations and Enhancement:

Pixel relationships: neighborhood, adjacency, and connectivity, Spatial domain fundamentals and convolution operation, Gray level transformations: log, power-law, contrast stretching, Histogram concept and histogram equalization, Spatial filtering: smoothing filters (linear and nonlinear)

MODULE 5 (8 hours)

Image Processing – Segmentation and Edge Detection:

Sharpening filters: gradient and Laplacian, Fundamentals of image segmentation, Thresholding: intensity and global thresholding, Region-based segmentation: region growing, splitting, and merging, Edge detection operators: Sobel and Prewitt

Text Books

1. Computer Graphics with OpenGL — Donald Hearn, M. Pauline Baker & Warren R. Carithers, 4th Edition.
2. Interactive Computer Graphics: A Top-Down Approach with WebGL/OpenGL — Edward Angel & Dave Shreiner, 6th Edition.
3. Computer Graphics: Principles and Practice — Foley, van Dam, Feiner & Hughes, 3rd Edition.
4. Fundamentals of Computer Graphics — Peter Shirley, Steve Marschner et al.
5. Rafael C. Gonzalez & Richard E. Woods, Digital Image Processing, 4th Edition, Pearson, 2018.
6. Real-Time Rendering — Tomas Akenine-Möller, Eric Haines, Naty Hoffman, Pearson, 4th Edition

Reference Books

1. James D. Foley, Andries van Dam, Steven Feiner, John F. Hughes, Computer Graphics: Principles and Practice, Addison-Wesley.
2. Zhigang Xiang & Roy A. Plastock, Computer Graphics (Schaum's Outline Series), McGraw-Hill.
3. William M. Newman & Robert F. Sproull, Principles of Interactive Computer Graphics, McGraw-Hill.
4. David F. Rogers, Procedural Elements for Computer Graphics, McGraw-Hill.
5. John F. Hughes et al., Computer Graphics: Principles and Practice in C, Addison-Wesley.
6. B. Chanda & D. Dutta Majumder, Digital Image Processing and Analysis, PHI Learning.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Computer Graphics - Foundations, Mathematics & 2D Graphics		10 hours
1.1	Overview of computer graphics systems, applications, components, and graphics pipeline	1
1.2	Coordinate systems, vectors, matrices, homogeneous coordinates, transformation composition	2
1.3	Raster graphics algorithms: Bresenham line drawing and midpoint circle algorithm	2
1.4	2D geometric transformations: translation, scaling, rotation	2
1.5	Windowing, viewport mapping, and 2D viewing pipeline	2
1.6	Introduction to Graphics API (OpenGL/WebGL)	1
Module 2: Computer Graphics – 3D Graphics, Viewing and Rendering		10 hours
2.1	3D transformations and matrix representation	2
2.2	Camera model and viewing pipeline	2
2.3	Orthographic and perspective projections	2
2.4	Polygonal modeling, vertex attributes, normals, and back-face culling	2
2.5	Hidden surface removal: Painter's algorithm and Z-buffer (conceptual)	2
2.6	Basic illumination and shading models (Flat, Gouraud, Phong – overview)	2
Module 3: Image Processing – Image Fundamentals and Representation		9 hours
3.1	Introduction to image processing and applications Image as 2D data	2
3.2	Binary, grayscale, and color image representation	2
3.3	Components of an image processing system	2
3.4	Fundamental steps in digital image processing Coordinate conventions	1
3.5	Sampling and quantization , Spatial and gray-level resolution	2

Module 4: : Image Processing – Spatial Domain Operations and Enhancemen		8 hours
4.1	Pixel relationships: neighborhood, adjacency, and connectivity	2
4.2	Spatial domain fundamentals and convolution operation	1
4.3	Gray level transformations: log, power-law, contrast stretching	2
4.4	Histogram concept and histogram equalization	1
4.5	Spatial filtering: smoothing filters (linear and nonlinear)	2
Module 5: : Image Processing – Segmentation and Edge Detection:		8 hours
5.1	Sharpening filters: gradient and Laplacian	2
5.2	Fundamentals of image segmentation	2
5.3	Thresholding: intensity and global thresholding	2
5.4	Region-based segmentation: region growing, splitting, and merging	1
5.5	Edge detection operators: Sobel and Prewitt	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain fundamentals of computer graphics systems, transformations, and rendering pipeline.
2. A simple drawing application allows users to draw lines and circles on the screen. Explain how a computer graphics system and graphics pipeline support this functionality.
3. A 2D object in a simulation needs to be moved and rotated smoothly on the screen. Explain how translation and rotation transformations are applied using homogeneous coordinates.
4. Compare raster and vector graphics with respect to their usage in real-world applications such as maps and digital drawings.

Course Outcome 2 (CO 2):

1. Apply viewing, modeling, and basic rendering techniques for 2D and 3D scenes.
2. A virtual camera is used in a 3D gaming environment. Explain the stages of the 3D viewing pipeline involved in displaying the scene on the screen.
3. A 3D object appears distorted when displayed. Explain how orthographic and perspective projections affect the appearance of objects in a graphics application.
4. In a 3D modeling application, hidden surfaces must not be displayed. Explain how the Z-buffer algorithm helps in solving this problem.

Course Outcome 3 (CO 3):

1. : Explain digital image representation, sampling, quantization, and pixel relationships
A medical image loses important details when zoomed in. Explain how sampling and quantization influence image quality in such applications.
2. Represent a given image as a 2D data structure and explain how binary, grayscale, and color images differ in storage and usage.
3. Explain pixel neighborhood, adjacency, and connectivity with reference to object detection in a digital image.

Course Outcome 4 (CO 4):

1. Apply spatial domain image enhancement and filtering techniques
2. A low-contrast satellite image needs to be enhanced for better interpretation. Explain how contrast stretching and histogram equalization can be applied.
3. A digital photograph is affected by salt-and-pepper noise. Suggest and justify a suitable spatial filtering technique to remove the noise.
4. Apply a 3×3 convolution mask to an image region and explain how spatial filtering modifies pixel values.

Course Outcome 5 (CO 5):

1. Apply image segmentation and edge detection techniques for image analysis
2. In a traffic surveillance system, vehicles must be separated from the background. Explain how global thresholding can be used for this purpose.
3. Explain region growing and region splitting & merging techniques with a suitable real-world image analysis scenario.
4. Compare Sobel and Prewitt edge detection operators and explain their role in identifying object boundaries in digital images.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CS3T09

Course Name: COMPUTER GRAPHICS AND IMAGE PROCESSING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. List any three applications of computer graphics.
2. What is homogeneous coordinate representation? State its advantage.
3. Define raster graphics.
4. What is perspective projection?
5. Define a digital image.
6. Differentiate between sampling and quantization
7. What is pixel connectivity?
8. Define histogram equalization.
9. What is image segmentation?
10. Name any two edge detection operators.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the major components of a computer graphics system with a neat block diagram. (7)
- (b) A 2D object is to be translated and rotated in a drawing application. Explain how 2D transformations are performed using homogeneous coordinates. (7)

OR

12. (a) Explain the graphics pipeline with reference to a simple drawing application (7)
- (b)) Rasterize a straight line using Bresenham's line drawing algorithm and explain its advantages (7)
13. (a) Explain 3D geometric transformations and their matrix representations (7)
- (b) Describe the stages of the 3D viewing pipeline used in a virtual reality environment. (7)
- i. Translate two unit in x-direction and three units in y-direction
- ii. Rotate the object by 45 degrees counter clockwise about the origin.

OR

14. (a) Differentiate between orthographic and perspective projections with suitable examples. (7)
- (b) Explain the Z-buffer algorithm used for hidden surface removal in 3D scenes. (7)
15. (a)) Explain how a digital image is represented as 2D data. Differentiate between binary, grayscale, and color images. (7)
- (b) Discuss the effect of sampling and quantization on image quality using a real-world imaging scenario. (7)

OR

16. (a) Explain the fundamental steps involved in digital image processing with a neat diagram. (7)
- (b) Explain pixel neighborhood, adjacency, and connectivity with suitable examples. (7)
17. (a) A low-contrast image captured under poor lighting conditions needs improvement. Explain suitable gray-level transformation techniques. (7)
- (b) Explain spatial domain filtering and the role of convolution in image processing (7)

OR

18. (a) Explain histogram equalization and its importance in image enhancement. (7)

p 1	0	0	2
0	4	6	1
6	0	1	3
6	1	q 0	4

- (b) Compare linear and non-linear spatial filters with suitable applications. (7)
19. (a) Explain global thresholding technique used for image segmentation with a suitable application. (7)
- (b) Describe region growing and region splitting & merging methods. (14)

3	2	1	1
4	4	5	5
5	5	6	7
1	2	6	7

OR

20. (a) Explain Sobel and Prewitt edge detection operators with suitable masks. (7)
- (b) Discuss the importance of edge detection in image analysis applications such as medical or satellite imaging. (7)

B24CS3P21	OBJECT MODELING AND DESIGN	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

Object Modeling and Design introduces the principles of object-oriented analysis and design for developing effective software systems. It focuses on identifying objects, classes, and their relationships to model real-world problems. The subject uses UML diagrams to represent system structure and behavior. It helps students design modular, reusable, and maintainable software solutions.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Understand object-oriented concepts and modeling principles, including objects, classes, associations, inheritance, and abstraction, and map them to real-world problems using class models. (Cognitive Knowledge Level: Understand)
CO 2	Construct advanced class models and state models using aggregation, inheritance, constraints, and state transitions to model system behavior. (Cognitive Knowledge Level: Apply)
CO 3	Develop advanced state and interaction models, including nested states, concurrency, use case, sequence, and activity diagrams, to describe system interactions. (Cognitive Knowledge Level: Apply)
CO 4	Analyze system requirements and domains by applying domain and application analysis techniques using class, state, and interaction models. (Cognitive Knowledge Level: Apply)
CO 5	Design software systems by applying system design principles, architectural styles, and subsystem decomposition to develop efficient and reusable solutions. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1		1		1	1	2	1		2
CO 2	3	2	1	1	2	1			1		
CO 3	2	1	2	2	2	1	1	3	3	2	1
CO 4	3	3	3	3	2	3	3	3	3	3	3
CO 5	2	3	3	3	3	3	2	2	2	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember			
Understand	40	40	40
Apply	60	60	60
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contains 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Introduction to OO and Modeling Concepts:

Introduction: What is Object Orientation?, What is OO development?, OO themes, Evidence for usefulness of OO development & Modeling as a Design Technique: Modeling, Abstraction, The three models. Class Modeling: Object and class concepts, Link and Associations Concepts, Generalization and Inheritance, A Sample Class Model, Navigation of class models, Practical tips.

MODULE 2 (7 hours)

Advanced Class Modeling and State Modeling:

Advanced Class Modeling: Advanced object and class concepts, Association ends, N-ary associations, Aggregation, Abstract classes, Multiple inheritance, Metadata, Reification, Constraints, Derived data, Packages, Practical tips. State Modeling: Events, States, Transitions and Conditions, State diagrams, State diagram behavior, Practical tips. *

MODULE 3 (8 hours)

Advanced State Modeling and Interaction Modeling:

Advanced State Modeling: Nested state diagrams, Nested states, Signal generalization, Concurrency, A sample state model, Relation of class and state models, Practical tips & Interaction Modeling: Use case models, Sequence models, Activity models. Advanced Interaction Modeling: Use case relationships; Special constructs for activity models. *

MODULE 4 (8 hours)

Analysis:

Process Overview: Development Stages, Development Life Cycle. System Conception: Devising a System Concept, Elaborating a Concept, Preparing a Problem Statement. Domain Analysis: Overview of Analysis, Domain Class Model, Domain State Model, Domain Interaction Model. Application Analysis: Application Interaction Model, Application Class Model, Application State Model. *

MODULE 5 (7 hours)

Design:

System Design: Overview of System Design, Estimating Performance, Making a Reuse Plan, Breaking a System into Subsystems, Identifying Concurrency, Allocation of Subsystems, Management of Data Storage, Handling Global Resources, Choosing a Software Control Strategy, Handling Boundary Conditions, Setting Trade-off Priorities, Common Architectural Styles, Architecture of the ATM System. *

(* UML Diagrams are used for various models)

Text Books

1. Object-Oriented Modeling and Design with UML - Michael Blaha, James Rumbaugh, 2nd Edition, Pearson Education, 2005.

Reference Books

1. Object-Oriented Analysis and Design with Applications - Grady Booch et al, 3rd Edition, Pearson, 2007
2. Practical Object-Oriented Design with UML – Mark Priestley, 2nd Edition, Tata McGraw-Hill, 2003.
3. The Unified Modeling Language User Guide – Booch, G., Rumbaugh, J., and Jacobson, I., 2nd Edition, Pearson, 2005.
4. Object-Oriented Systems Analysis and Design Using UML – Simon Bennett, Steve McRobb and Ray Farmer, 2nd Edition, Tata McGraw-Hill, 2002.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to OO and Modelling Concepts:		6 hours
1.1	Introduction: What is Object Orientation?, What is OO development? OO themes, Evidence for usefulness of OO development	1
1.2	Modeling as Design Technique: Modeling, abstraction, The three models.	1
1.3	Class Modeling: Object and class concepts	1

1.4	Link and associations concepts	1
1.5	Generalization and inheritance	1
1.6	A sample class model, Navigation of class models, Practical tips	1
Module 2: Advanced Class Modelling and State Modelling:		7 hours
2.1	Advanced Class Modelling: Advanced object and class concepts, Association ends, N-ary associations	1
2.2	Aggregation, Abstract classes	1
2.3	Multiple inheritance	1
2.4	Metadata, Reification, Constraints, Derived data	1
2.5	Packages, Practical tips. State Modeling: Events, States,	1
2.6	Transitions and Conditions, State diagrams	1
2.7	State diagram behavior, Practical tips.	1
Module 3: Advanced State Modelling and Interaction Modelling:		8 hours
3.1	Advanced State Modeling: Nested state diagrams, Nested states	1
3.2	Signal generalization,Concurrency	1
3.3	A sample state model, Relation of class and state models, Practical tips.	1
3.4	Interaction Modeling: Use case models	1
3.5	Sequence models	1
3.6	Activity models	1
3.7	Advanced Interaction Modelling: Use case relationships	1
3.8	Special constructs for activity models.	1
Module 4: Analysis:		8 hours
4.1	Process Overview: Development Stages,Development Life Cycle.	1
4.2	System Conception: Devising a System Concept, Elaborating a Concept, Preparing a Problem Statement.	1
4.3	Domain Analysis: Overview of Analysis, Domain Class Model,	1
4.4	Domain Class Model	1
4.5	Domain State Model, Domain Interaction Model	1

4.6	Application Analysis: Application Interaction Model	1
4.7	Application Class Model	1
4.8	Application State Model	1
Module 5: Design:		7 hours
5.1	System Design: Overview of System Design, Estimating Performance, Making a Reuse Plan	1
5.2	Breaking a System into Subsystems, Identifying Concurrency	1
5.3	Allocation of Subsystems, Management of Data Storage	1
5.4	Handling Global Resources, Choosing a Software Control Strategy, Handling Boundary Conditions, Setting Trade-off Priorities	2
5.5	Common Architectural Styles	1
5.6	Architecture of the ATM System.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Illustrate the usefulness of Object Oriented Design.
2. Demonstrate the significance of generalization and inheritance in Class Modeling.

Course Outcome 2 (CO 2):

1. Illustrate the role of abstract classes and constraints with the help of an example.
2. An elevator in a multi-storey building operates in the states Idle, Moving Up, Moving Down, Door Open, Door Closed, and Emergency Stop. Events such as floor requests, arrival at a floor, door operations, and emergency activation cause state changes. Construct a UML State Diagram for the elevator system showing the states, transitions, triggering events, guard conditions (if any), and the system behavior during emergency situation

Course Outcome 3 (CO 3):

1. A washing machine controller has a state called Washing Cycle, which contains the substates Soaking, Washing, and Rinsing. Explain the concepts of nested states and nested state diagrams. Illustrate how the Washing Cycle can be represented using a UML State Diagram and discuss the advantages of state hierarchy.

2. A university library management system allows students to search for books, borrow available books, return borrowed books, and reserve books that are currently unavailable. The system maintains information about students, books, reservations, and issue transactions. Based on the given scenario, develop suitable UML interaction models, including a Use Case Diagram showing the actors and use cases, a Sequence Diagram for the book borrowing process, and an Activity Diagram representing the workflow from book search to successful issue of a book. Clearly indicate the interactions, decision points, and system responses involved in the process.

Course Outcome 4 (CO 4):

1. A municipal corporation intends to develop a Smart Waste Management System to monitor garbage collection and disposal activities. Explain the process of devising a system concept, elaborating the concept, and preparing a problem statement. Illustrate how these activities contribute to the successful development of the Smart Waste Management System.
2. A city plans to implement a Smart Parking Management System that enables drivers to locate available parking spaces, reserve parking slots, make online payments, and receive notifications regarding parking availability and reservation status. Parking administrators are responsible for managing parking zones, monitoring occupancy, and maintaining parking records. Based on the given scenario, prepare a suitable problem statement and develop the necessary object-oriented analysis and design models. Construct a Domain Class Model identifying the key entities and their relationships, a Domain Interaction Model representing the parking reservation process, and an Application Class Model incorporating system-specific classes, attributes, and operations. Also, explain how these models support the various stages of the object-oriented development life cycle.

Course Outcome 5 (CO 5):

1. A cloud-based document management system stores files for thousands of users and provides shared access to documents. Explain the challenges associated with data storage and handling global resources in system design. Discuss suitable strategies for ensuring data consistency, reliability, and controlled access to shared resources.
2. An e-commerce platform enables customers to browse products, place orders, make online payments, track deliveries, and submit product reviews. The system must support a large number of concurrent users, especially during promotional events and seasonal sales. Various services such as inventory management, payment processing, order fulfillment, and customer support must operate efficiently and reliably. Based on the given scenario, apply system design principles to identify suitable subsystems, determine shared global resources and methods for managing them, recommend an appropriate architectural style, and propose mechanisms for handling exceptional conditions such as payment failures and inventory shortages. Also, discuss how the allocation of subsystems can improve the overall performance, scalability, and maintainability of the platform.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CS3P21

Course Name: OBJECT MODELING AND DESIGN

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Illustrate the usefulness of Object Oriented Development.
2. Demonstrate link attributes with examples.
3. Create a simple state diagram for a mobile phone's battery, showing the states: Charging, Discharging, and Fully Charged. Label each transition with the corresponding event (e.g., plug in, unplug).
4. Illustrate with an example how abstract classes and multiple inheritance can be used in a payroll system to model different types of employees (e.g., salaried, hourly, contract).
5. Describe how concurrency is represented in a state diagram for a multi-user online banking system. Use a simple example to show concurrent states.
6. Describe the difference between "include" and "extend" relationships in use case modeling. Give one example for each.
7. Consider a Food Ordering Application. Apply application analysis concepts to describe
 - (a) one application interaction
 - (b) one application class
 - (c) one application state that would be involved when a user places an order

8. Explain the objectives of application analysis. How do the application interaction model, application class model, and application state model contribute to understanding system behavior?
9. Describe the importance of identifying concurrency in a system.
10. Identify and describe any two subsystems for an online ticket booking system, by applying system design concepts.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. Describe the various stages involved in the object oriented methodology. (14)

OR

12. Illustrate the generalization and inheritance features of object oriented modeling with the help of examples. (14)
13. A smart home system controls a door lock that can be in one of four states: Locked, Unlocked, Alarm, and Maintenance. The system responds to the following events: "Unlock Request," "Lock Request," "Intrusion Detected," and "Maintenance Mode Request." Each transition between states is governed by specific conditions, such as the correct password for unlocking or an authorized technician for maintenance mode. The alarm state is triggered if an intrusion is detected while the door is locked. Draw a state diagram for the door lock, showing all states, transitions, and the events and conditions that govern each transition. (14)

OR

14. (a) Illustrate Entry and Exit Actions with examples. (7)
(b) Illustrate Internal Actions with examples. (7)
15. For a university course registration system, create a use case model showing actors and their interactions. Then, develop a sequence diagram for the registration process, illustrating interactions between the student, registration system, and database. Explain how the sequence model supports the use case model and describe any special constructs used in the activity model for handling exceptions. (14)

OR

16. Design a nested state diagram for a smart washing machine that includes main states like "Idle," "Washing," "Rinsing," "Spinning," and "Error." Each main state should have relevant substates (e.g., "Washing" may have "Filling," "Agitating," "Draining"). Explain how nested states help manage complexity and show the relationship between the class and state models for this system. (14)
17. A university plans to develop an Online Course Registration System. Apply the object-oriented development life cycle to this system by: (14)

- i. Identifying and explaining each development stage.
- ii. Describing the activities performed in each stage.
- iii. Explaining how iteration and feedback improve system quality.

OR

18. (a) Explain the differences between domain analysis and application analysis. Compare them with respect to their objectives, models used, and outcomes in object-oriented system design. (7)
- (b) Explain the process overview of object-oriented software development. Describe the major development stages and explain how they together form a development life cycle. (7)
19. (a) Explain the steps involved in breaking a system into subsystems and discuss how subsystem allocation is performed. (7)
- (b) A mobile payment application must balance performance, security, and scalability. Apply trade-off prioritization and explain how design decisions are made to resolve conflicts among these requirements. (7)

OR

20. (a) Explain how data storage management and handling global resources are addressed during system design. (7)
- (b) A company plans to develop an Online Shopping System. Apply system design principles to:
- i. break the system into subsystems
 - ii. allocate responsibilities to each subsystem.
- (7)

B24CS3P22A	CRYPTOGRAPHY AND NETWORK SECURITY (A)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4		2024

Preamble

This course aims to provide an in-depth understanding of modern cryptographic techniques and network security mechanisms used to protect data and communication systems. It covers pseudorandomness, key management, authentication systems, cloud and wireless security, e-mail security, and firewall technologies. The knowledge gained from this course will help students design, analyze, and apply security solutions in advanced computing environments and future cybersecurity-related roles.

Prerequisites

Basic knowledge of computer networks, data communication, and fundamental mathematical concepts such as modular arithmetic and number systems.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain computer security concepts, attack models, and block cipher design principles. (Cognitive Knowledge Level: Understand)
CO 2	Explain pseudorandom number generation and basic cryptographic primitives. (Cognitive Knowledge Level: Understand)
CO 3	Apply key management techniques and user authentication mechanisms. (Cognitive Knowledge Level: Apply)
CO 4	Analyze security risks and protection mechanisms in cloud and wireless networks. (Cognitive Knowledge Level: Analyze)
CO 5	Apply e-mail and perimeter security mechanisms for network protection. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	2	2	2	1			1		3
CO 2	3	2	1	2	1				1		3
CO 3	2	3	3	3	3	3	2		1		3
CO 4	2	3	3	3	3	3	2		1		3
CO 5	2	3	3	3	3	3	2		1		3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	50	50	50
Apply	20	20	20
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Foundations of Modern Cryptography:

Computer Security Concepts: Confidentiality, Integrity, Availability, Authenticity, Accountability. Types of Attacks. A model for Network Security. Cryptography, Cryptanalysis and Brute-Force Attack. Block cipher design principles (Basic Concepts).

MODULE 2 (8 hours)

Pseudorandomness and Cryptographic Primitives:

Principles of Pseudorandom number generation, Pseudorandom number generators, Pseudorandom number generation using a Block Cipher. Symmetric Encryption System, Public Key Cryptography, Message Authentication and Hash functions (Basic Concepts).

MODULE 3 (6 hours)

Key Management and Authentication Systems:

Key Management and Distribution: Key distribution using symmetric and asymmetric encryption, Distribution of public keys, X.509 certificates, Public-Key Infrastructure. User Authentication: User authentication principles, Kerberos V4, Federated Identity Management.

MODULE 4 (7 hours)

Cloud and Wireless Network Security:

Network Access control, Cloud computing, Cloud security Risks and Countermeasures, Data Protection in the cloud, Cloud security as a service. Wireless security, IEEE 802.11i Wireless LAN Security.

MODULE 5 (8 hours)

E-mail and Perimeter Network Security:

Internet mail architecture, Email formats, E-mail threats, S/MIME, PGP. Firewalls: Design Principles, Trusted Systems, Packet filtering, Stateful inspection, Application gateways.

Text Books

1. William Stallings, Cryptography and Network Security: Principles and Practice, Pearson.
2. Atul Kahate, Cryptography and Network Security, McGraw Hill.
3. Christof Paar and Jan Pelzl, Understanding Cryptography: A Textbook for Students and Practitioners, Springer.

Reference Books

1. Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone, Handbook of Applied Cryptography, CRC Press.
2. Charlie Kaufman, Radia Perlman, and Mike Speciner, Network Security: Private Communication in a Public World, Prentice Hall.
3. Bernard Menezes, Network Security and Cryptography, Cengage Learning.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Foundations of Modern Cryptography		7 hours
1.1	Computer Security Concepts: Confidentiality, Integrity, Availability	1
1.2	Authenticity and Accountability, Relationship between security services and mechanisms	1
1.3	Types of Attacks: Passive, Active, Insider, Outsider attacks	1
1.4	Model for Network Security (sender, receiver, adversary, trusted third party)	1
1.5	Cryptography vs Cryptanalysis, Brute-Force Attack: Concept and limitations	1
1.6	Introduction to Block Ciphers, Need for block cipher design principles	1

1.7	Block Cipher Design Principles: Confusion, Diffusion, Avalanche effect (basic concepts)	1
Module 2: Pseudorandomness and Cryptographic Primitives		8 hours
2.1	Need for randomness in cryptography, Principles of pseudorandom number generation	1
2.2	Pseudorandom Number Generators (PRNG): Properties and evaluation criteria	1
2.3	Pseudorandom number generation using block ciphers (conceptual working)	1
2.4	Overview of Cryptographic Primitives, Symmetric Encryption System: Components and operation	1
2.5	Public Key Cryptography: Basic concepts and comparison with symmetric systems	1
2.6	Message Authentication: Goals and basic mechanisms	1
2.7	Hash Functions: Purpose, properties, applications	1
2.8	Integration of primitives: Encryption + Authentication + Hashing in secure systems	1
Module 3: Key Management and Authentication Systems		6 hours
3.1	Importance of Key Management, Key distribution using symmetric encryption	1
3.2	Key distribution using asymmetric encryption	1
3.3	Distribution of Public Keys, X.509 Certificates: Structure and purpose	1
3.4	Public Key Infrastructure (PKI): Components and operation	1
3.5	User Authentication Principles, Password-based and token-based authentication	1
3.6	Kerberos V4: Working principle, Federated Identity Management: Concept and benefits	1
Module 4: Cloud and Wireless Network Security		7 hours
4.1	Network Access Control (NAC): Need and mechanisms	1
4.2	Cloud Computing: Service models and deployment models	1
4.3	Cloud Security Risks: Data breaches, multi-tenancy, insider threats	1
4.4	Cloud Security Countermeasures	1

4.5	Data Protection in the Cloud, Encryption at rest and in transit	1
4.6	Cloud Security as a Service (SECaaS)	1
4.7	Wireless Security Overview, IEEE 802.11i Wireless LAN Security Architecture	1
Module 5: E-mail and Perimeter Network Security		8 hours
5.1	Internet Mail Architecture, Mail transfer and delivery agents	1
5.2	Email Formats and Standards	1
5.3	Email Threats: Spoofing, Phishing, Malware, Spam	1
5.4	S/MIME: Architecture and services	1
5.5	PGP: Working principles and comparison with S/MIME	1
5.6	Firewall Design Principles, Trusted Systems	1
5.7	Firewall Types: Packet Filtering and Stateful Inspection	1
5.8	Application Gateways (Proxy Firewalls), Firewall deployment scenarios	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the core computer security concepts such as confidentiality, integrity, availability, and accountability with suitable examples.
2. Describe different types of security attacks and explain the network security model used to counter these attacks.

Course Outcome 2 (CO 2):

1. Explain the principles of pseudorandom number generation and discuss the properties of a good pseudorandom number generator.
2. Describe the basic concepts of symmetric encryption, public key cryptography, message authentication, and hash functions.

Course Outcome 3 (CO 3):

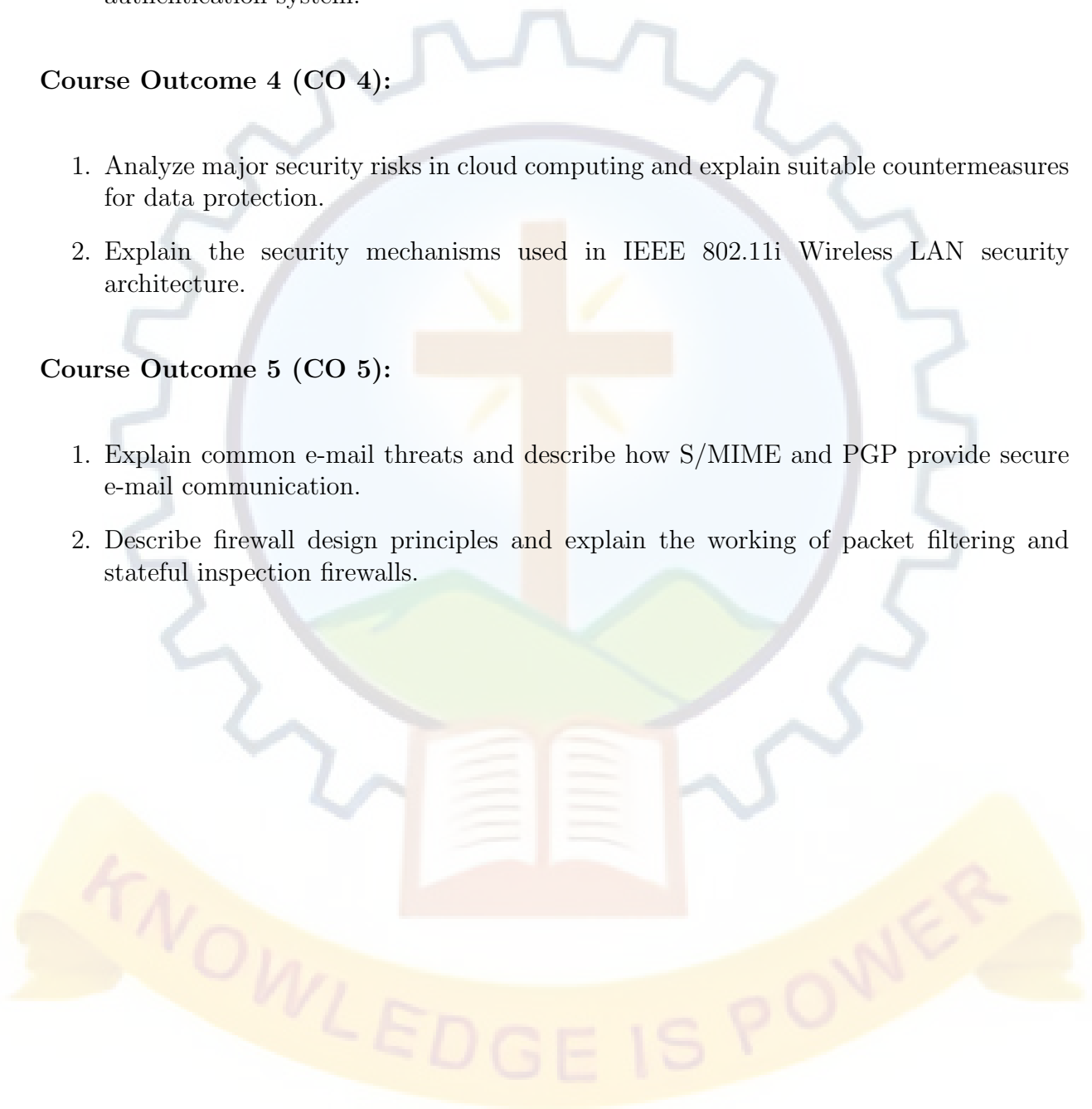
1. Explain key distribution mechanisms using symmetric and asymmetric encryption techniques.
2. Describe the working principles of Public Key Infrastructure (PKI) and Kerberos authentication system.

Course Outcome 4 (CO 4):

1. Analyze major security risks in cloud computing and explain suitable countermeasures for data protection.
2. Explain the security mechanisms used in IEEE 802.11i Wireless LAN security architecture.

Course Outcome 5 (CO 5):

1. Explain common e-mail threats and describe how S/MIME and PGP provide secure e-mail communication.
2. Describe firewall design principles and explain the working of packet filtering and stateful inspection firewalls.



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CS3P22A

Course Name: CRYPTOGRAPHY AND NETWORK SECURITY (A)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define confidentiality, integrity, and availability in the context of computer security.
2. Distinguish between cryptography and cryptanalysis.
3. What is a pseudorandom number generator? List the properties of a good PRNG.
4. Define cryptographic primitives and list any four basic primitives used in security systems.
5. What is the need for key management in cryptographic systems?
6. What is an X.509 certificate? Mention its purpose.
7. List any three major security risks associated with cloud computing.
8. What is the role of IEEE 802.11i in wireless LAN security?
9. List common e-mail security threats.
10. What is a firewall? State its primary functions.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain different types of security attacks with suitable examples. (7)
(b) Describe the network security model and explain its components. (7)

OR

12. (a) Explain brute-force attacks and their impact on cryptographic systems. (6)
(b) Discuss basic block cipher design principles such as confusion, diffusion, and avalanche effect. (8)
13. (a) Explain the principles of pseudorandom number generation in cryptographic systems. (7)
(b) Describe pseudorandom number generation using a block cipher. (7)

OR

14. (a) Explain the need for cryptographic primitives in secure communication. (5)
(b) Describe the basic concepts of symmetric encryption, public key cryptography, message authentication, and hash functions. (9)
15. (a) Explain key distribution using symmetric encryption. (6)
(b) Describe key distribution using asymmetric encryption techniques. (8)

OR

16. (a) Explain the role and components of Public Key Infrastructure (PKI). (7)
(b) Explain the role and components of Public Key Infrastructure (PKI). (7)
17. (a) Explain network access control mechanisms. (5)
(b) Discuss cloud security risks and corresponding countermeasures. (9)

OR

18. (a) Explain data protection mechanisms used in cloud environments. (7)
(b) Describe the IEEE 802.11i wireless LAN security architecture. (7)
19. (a) Explain the Internet mail architecture and standard e-mail formats. (6)
(b) Discuss common e-mail threats and their impact. (8)

OR

20. (a) Explain S/MIME and PGP mechanisms for secure e-mail communication. (7)
(b) Describe firewall design principles and explain different types of firewalls. (7)

B24CS3P23	LAN TECHNOLOGIES	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2025

Preamble

This course provides an overview of Local Area Network (LAN) technologies, covering network architectures, physical and data-link layer mechanisms, LAN standards, and access methods. It also introduces key internetworking components, network management principles, Wide Area Networks (WANs), and fundamental network security concepts essential for modern enterprise networking.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Describe the fundamental concepts of LANs, including network architectures, protocols, topologies, and physical-layer technologies.(Cognitive Knowledge Level: Understand)
CO 2	Explain data communication techniques such as signal encoding, error detection and correction, data link control, and LAN access mechanisms.(Cognitive Knowledge Level: Understand)
CO 3	Analyze the characteristics and functioning of major LAN technologies including Ethernet, Token Passing LANs, ATM LANs, and Wireless LANs. (Cognitive Knowledge Level: Apply)
CO 4	Examine internetworking devices and protocols, and evaluate network management approaches and tools.(Cognitive Knowledge Level: Apply)
CO 5	Discuss WAN technologies and apply fundamental security principles, including cryptography, firewalls, and VPN mechanisms, in network environments.(Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	1									
CO2	3	2	1	3	2						
CO3	2	3	3	3	3	3					
CO4	2	3	3	3	3	3	2				
CO5	2	3	3	3	3	3	3				

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember			
Understand	40	40	40
Apply	60	60	60
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Introduction to LANs and Physical Layer Technologies:

Local Area Network introduction – OSI / TCP-IP layered architecture – Types of networks – Basic LAN terminology – LAN standards – LAN features and topologies – LAN protocols – LAN components. Applications of LANs.

MODULE 2 (7 hours)

Data Communication Concepts and LAN Access Methods:

Signal encoding techniques – Error detection – Error correction – Data link control – Transmission media – Multiplexing. LAN access techniques: Random access, controlled access schemes, switched access methods.

MODULE 3 (7 hours)

Ethernet, Token Ring, Wireless LANs:

Ethernet: standard Ethernet, fast Ethernet. Token passing LANs: classic, dedicated, high-speed LAN. Wireless LANs: overview, MAC layer, spread-spectrum wireless LAN, infrared wireless LAN. LAN databases: database service and server-based applications.

MODULE 4 (8 hours)

Internetworking and Network Management:

Internetworking perspective – Bridges – Routers – Internetworking switches – Virtual LANs – Spanning tree protocol. Network management: functions, elements of network management, centralized vs decentralized management, network management protocols (SNMP, RMON, SMON).

MODULE 5 (8 hours)

Wide Area Networks and Network Security:

Wide Area Networks: private networks, public networks, circuit switching and packet switching networks, interconnection of public and private data networks, value added networks, WAN applications. Network security: basic security issues, security policies, cryptography, firewalls, IP security, virtual private networks.

Text Books

1. Local Area Networks, Tata McGraw Hill Edition, Gerd Keiser
2. Local Area Networks, S. K. Basandra, S. Jaiswal
3. Local Area Networks: An introduction to the Technology, John E McNamara

Reference Books

1. Local Area Networks: An introduction to the Technology, John E McNamara

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	Hours
Total Hours		36 Hours
Module 1: Local Area Network Introduction		6 hours
1.1	OSI / TCP-IP layered architecture and Types of Networks	2
1.2	Basic LAN terminology, LAN standards	1
1.3	LAN features, LAN topologies	1
1.4	LAN components, LAN protocols	1
1.5	Applications of LANs	1
Module 2: Data Communication Concepts and LAN Access		7 hours
2.1	Signal Encoding Techniques	1
2.2	Error Detection and Error Correction	1
2.3	Data Link Control	1
2.4	Transmission Media (detailed)	1
2.5	Multiplexing Techniques	1
2.6	LAN Access techniques: Random Access and Controlled Access Schemes	1
2.7	Switched Access Methods	1
Module 3: Ethernet, Token Passing LANs, Wireless LANs		7 hours
3.1	Standard Ethernet	1

3.2	Fast Ethernet	1
3.3	Token Passing LANs: Classic	1
3.4	Dedicated and High-Speed Token Ring LANs	1
3.5	Wireless LANs: Overview and Wireless LAN MAC Layer	1
3.6	Spread-Spectrum and Infrared Wireless LANs	1
3.7	LAN Databases: Database service	1
Module 4: Internetworking and Network Management		8 hours
4.1	Internetworking perspective	1
4.2	Bridges and Routers	1
4.3	Internetworking switches, VLANs	1
4.4	Spanning Tree Protocol (STP)	1
4.5	Network Management Functions	1
4.6	Network Management Elements	1
4.7	Centralized vs Decentralized Management	1
4.8	Network Management Protocols: SNMP, RMON, SMON	1
Module 5: Wide Area Networks and Network Security		8 hours
5.1	WANs: Private Networks and Public Networks	1
5.2	Circuit Switching, Packet Switching	1
5.3	Interconnection of public & private data networks	1
5.4	Value Added Networks, WAN Applications	1
5.5	Basic Security Issues and Security Policies	1
5.6	Cryptography and Firewalls	1
5.7	IP Security (IPSec)	1
5.8	Virtual Private Networks (VPNs)	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the features and advantages of Local Area Networks. Distinguish between different LAN topologies with examples.

2. Describe the OSI and TCP/IP layered architectures. Compare their functionalities and discuss how LAN protocols operate within these layers.

Course Outcome 2 (CO2):

1. Explain error detection and correction techniques with suitable diagrams. Compare CRC and Hamming Code.
2. Analyze different LAN access mechanisms such as ALOHA, CSMA/CD, and controlled access schemes. Discuss their efficiency and collision behavior.

Course Outcome 3 (CO3):

1. Compare Standard Ethernet and Fast Ethernet in terms of speed, operation, and frame structure.
2. Discuss the architecture and working of Wireless LANs. Explain the role of the MAC layer and spread-spectrum techniques.

Course Outcome 4 (CO4):

1. Explain the functions of bridges, routers, and switches in LAN internetworking. How does VLAN segmentation improve network performance?
2. Describe the purpose of Spanning Tree Protocol (STP). Explain how loops are prevented in switched networks.

Course Outcome 5 (CO5):

1. Distinguish between circuit switching and packet switching. Discuss their applications in WANs.
2. Explain the role of firewalls, cryptography, and VPNs in securing network communication. Describe the working of IPSec.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CS3P23

Course Name: LAN TECHNOLOGIES

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define a Local Area Network (LAN). List any two features of LAN.
2. Distinguish between OSI and TCP/IP reference models.
3. List any three LAN topologies and mention one advantage of each.
4. Define transmission media. Classify transmission media used in LANs.
5. What is signal encoding? Mention any two encoding techniques.
6. Define error detection and error correction.
7. What is multiplexing? Name any two multiplexing techniques.
8. What is CSMA/CD? Where is it used?
9. What is the function of a network bridge?
10. List any three network security mechanisms.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the layered architecture of the OSI model with neat diagram. (7)
(b) Describe different types of networks with suitable examples. (7)

OR

12. (a) Explain LAN features and various LAN topologies. (7)
(b) Discuss LAN standards and basic LAN terminology. (7)
13. (a) Explain different signal encoding techniques used in data communication. (7)
(b) Describe error detection and correction techniques. (7)

OR

14. (a) Explain multiplexing techniques with suitable examples. (7)
(b) Describe ALOHA and CSMA/CD access methods. (7)
15. (a) Explain the architecture and operation of standard Ethernet. (7)
(b) Discuss Token Ring LAN and its operation. (7)

OR

16. (a) Spread-spectrum wireless LAN – concept and techniques (7)
(b) Infrared wireless LAN – working principle, characteristics, and applications (7)
17. (a) Explain internetworking concepts and functions of routers. (7)
(b) Describe Virtual LANs and Spanning Tree Protocol. (7)

OR

18. (a) Explain network management functions and elements. (7)
(b) Discuss SNMP and RMON protocols. (7)
19. (a) Explain circuit switching and packet switching networks. (7)
(b) Discuss WAN applications and value added networks. (7)

OR

20. (a) Explain basic network security issues and security policies. (7)
(b) Describe firewalls, VPNs and IP security. (7)

B24CS3P24	WEB MINING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION 2024
		2	1	0	4	3	

Preamble

This course provides a foundation in web mining, covering key concepts in information retrieval, structured data extraction, and web search, with a special emphasis on web crawling. Students will learn to apply various techniques in web usage mining to solve real-world problems.

Prerequisites

B24CS2T06 Database management systems

Course Outcomes

After the completion of the course, the student will be able to:

CO 1	Analyze the core components of web mining and evaluate their interrelationships in building effective web-based systems.(Cognitive Knowledge Level: Apply)
CO 2	Apply web mining algorithms to analyze networks and discover patterns.(Cognitive Knowledge Level: Apply)
CO 3	Design and evaluate various web crawling architectures while assessing the implementation issues, ethical considerations, and performance metrics associated with large-scale web crawling. (Cognitive Knowledge Level: Understand)
CO 4	Develop the role of Structured Data Extraction in web structure mining. (Cognitive Knowledge Level: Apply)
CO 5	Implement a complete web usage mining process, from data collection to pattern analysis, for practical application.(Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	1	2	1				1		2
CO 2	3	3	2	3	3	1		1	3	1	3
CO 3	2	3	3	3	3	3	3	3	3	3	3
CO 4	2	2	2	3	2	1		1	2	1	2
CO 5	3	3	3	3	3	3	2	2	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction to web mining

Introduction – Web Mining – Theoretical background – Association rule mining – ECLAT algorithm, Sequential Pattern Mining- Generalized Sequential Pattern algorithm, Information retrieval and Web search, Information retrieval Models- Boolean retrieval model, Vector space model- Document Representation, Retrieval and Relevance Ranking, probabilistic model, Relevance Feedback, Text and Web page Pre-processing, Inverted Index , Web Search – Meta-Search – Web Spamming

MODULE 2 (7 hours)

Social Networks Analysis

Introduction -Social Networks Analysis- Centrality and Prestige ,Co-Citation and Bibliographic Coupling- Page Rank: PageRank Algorithm, Link-Based Similarity Search, Enhanced Techniques for Page Ranking - HITS: HITS Algorithm, Finding Other Eigenvectors

MODULE 3 (7 hours)

Web Crawling

Web Crawling Web Crawling -A Basic Crawler Algorithm: Breadth-First Crawlers, Preferential Crawlers - Implementation Issues, Universal Crawlers- Focused Crawlers, Topical Crawlers, Crawler Ethics and Conflicts

MODULE 4 (8 hours)

Structured Data Extraction

Structured Data Extraction: Wrapper Generation – Preliminaries- Wrapper Induction- Instance Based Wrapper Learning, Automatic Wrapper Generation: Issues, String Matching Techniques, and Tree Matching Approaches, Multiple Alignment - Building DOM Trees, Extraction Based on a Single List Page and Multiple pages, Introduction to Schema Matching – Schema Level Match -Domain and Instance Level Matching – Extracting and Analysing Web Social Networks.

MODULE 5 (7 hours)

Web Usage Mining

Web Usage Mining - Data Collection and Pre-Processing: Sources and Types of Data, Key Elements of Web Usage Data ,Data Modelling for Web Usage Mining - Discovery and Analysis of Web Usage Patterns – Applications- Recommender Systems and Collaborative Filtering – Query Log Mining.

Text Books

1. Bing Liu, “Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data (Data-Centric Systems and Applications)”, Springer; 2nd Edition, 2009.

Reference Books

1. Zdravko Markov and Daniel T. Larose, “Data Mining the Web: Uncovering Patterns in Web Content, Structure, and Usage”, John Wiley & Sons, Inc., 2007.
2. Guandong Xu, Yanchun Zhang, and Lin Li, “Web Mining and Social Networking: Techniques and Applications”, Springer; 1st Edition, 2010.
3. Soumen Chakrabarti, “Mining the Web: Discovering Knowledge from Hypertext Data”, Morgan Kaufmann; Edition, 2002.
4. Adam Schenker, “Graph-Theoretic Techniques for Web Content Mining”, World Scientific Publishing Co. Inc., 2005.
5. Min Song, Yi Fang, and Brook Wu, “Handbook of Research on Text and Web Mining Technologies”, IGI Global, Information Science Reference – imprint of IGI Publishing, 2008.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Web Mining – Theoretical Background		7 hours
1.1	Introduction to Web Mining	1
1.2	Association Rule Mining – Concepts	1
1.3	ECLAT Algorithm	1
1.4	Sequential Pattern Mining	1
1.5	Generalized Sequential Pattern Algorithm	1
1.6	IR Models – Boolean, VSM, Probabilistic	1
1.7	Relevance Feedback, Pre-processing, Inverted Index, Web Search	1

Module 2: Social Network Analysis		7 hours
2.1	Introduction to Social Network Analysis	1
2.2	Centrality and Prestige Measures	1
2.3	Co-Citation and Bibliographic Coupling	1
2.4	PageRank – Algorithm and Applications	1
2.5	Link-Based Similarity Search	1
2.6	Enhanced Page Ranking Techniques	1
2.7	HITS Algorithm – Hub and Authority Scores	1
Module 3: Web Crawling		7 hours
3.1	Introduction to Web Crawling	1
3.2	Basic Crawler Algorithm – Breadth-First Crawlers	1
3.3	Preferential Crawlers – Concepts and Design	1
3.4	Crawler Implementation Issues	1
3.5	Universal Crawlers	1
3.6	Focused and Topical Crawlers	1
3.7	Crawler Ethics and Conflicts	1
Module 4: Structured Data Extraction		8 hours
4.1	Wrapper Generation – Preliminaries	1
4.2	Wrapper Induction – Instance-Based Wrapper Learning	1
4.3	Automatic Wrapper Generation – String Matching Techniques	1
4.4	Tree Matching Approaches	1
4.5	Multiple Alignment and DOM Tree Construction	1
4.6	Extraction from Single and Multiple List Pages	1
4.7	Introduction to Schema Matching – Domain and Instance Level Matching	1
4.8	Extracting and Analysing Web Social Networks	1
Module 5: Web Usage Mining		7 hours
5.1	Web Usage Mining – Data Collection	1

5.2	Data Preprocessing – Sources and Types of Data	1
5.3	Modelling Web Usage Data – Key Elements	1
5.4	Discovery and Analysis of Web Usage Patterns	1
5.5	Recommender Systems – Concepts	1
5.6	Collaborative Filtering Techniques	1
5.7	Query Log Mining – Applications	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Illustrate Latent Semantic Indexing.
2. Discuss various applications of Web Mining in real-world scenarios.

Course Outcome 2 (CO2):

1. Explain Naïve Bayesian Text Classification
2. Compare supervised, unsupervised, and semi-supervised learning approaches.

Course Outcome 3 (CO3):

1. Make use of an example to explain the working principle of Hyperlink based Ranking.
2. Describe Link-Based Similarity Search in page ranking.
3. Discuss the importance of PageRank and HITS algorithms in Web Search.

Course Outcome 4 (CO4):

1. Discuss Extracting and Analyzing Web Social Networks.
2. Explain the role of structure-based data extraction in Web Mining.

Course Outcome 5 (CO5):

1. Explain Probabilistic Latent Semantic Analysis.
2. Discuss different techniques for Web Usage Pattern Analysis.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CS3P24

Course Name: WEB MINING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Differentiate between Information retrieval and Web search in web mining.
2. Explain Boolean model for Information Retrieval System.
3. Describe the role of link analysis and random walks in the PageRank algorithm.
4. Discuss the differences between co-citation and bibliographic coupling as network analysis techniques.
5. Compare breadth-first crawlers and preferential crawlers in terms of their crawling strategies and efficiency.
6. Explain the concept of focused crawling based on social networks or user behaviour.
7. List out the techniques used in string matching and tree matching for automatic wrapper generation.
8. Summarize the process of analysing web social networks.
9. Explain Web Recommender systems based on User and Item.
10. Describe Data Modelling for Web Usage Mining and its role in representing and analysing user behaviour.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the characteristics of Web search. (6)
- (b) Use Latent Semantic Indexing (LSI) to rank these documents for the query “gold silver truck” for the following documents: (8)
- d1: Shipment of gold damaged in a fire.
 - d2: Delivery of silver arrived in a silver truck.
 - d3: Shipment of gold arrived in a truck.

OR

12. An e-commerce website maintains a large transaction database containing records of items purchased together by users. The website aims to discover frequently co-occurring products in order to generate effective product recommendations. Since the dataset is dense and contains repeated item combinations, the traditional Apriori approach becomes inefficient.
- a) Explain how the ECLAT algorithm can be applied to this problem to mine frequent itemsets.
- b) Describe how the transaction data is represented, how frequent itemsets are generated and their support is computed,
- c) Why ECLAT is suitable for this scenario. Also discuss any limitations of using ECLAT in large-scale web-based applications.
13. (a) Illustrate the concept of Link-Based Similarity Search in Page Ranking. (6)
- (b) Describe how the HITS algorithm can be extended to find other eigenvectors beyond the top-ranked ones. (8)

OR

14. (a) Differentiate Schema Level Match-Domain and Instance Level Matching in Web mining. (8)
- (b) Describe the problem of community discovery in social networks and explain its significance. (6)
15. (a) Discuss potential techniques to improve the performance and effectiveness of basic crawler algorithms. (8)
- (b) Illustrate the evaluation methods used to assess the performance and quality of topical crawlers. (6)

OR

16. Describe the approaches and techniques used to address ethical and conflict-related challenges in web crawling. (14)
17. Prepare Document Object Model (DOM) tree from the given HTML page. (14)

```
1 <html>
2 <head>
3   <title>Here comes the DOM</title>
4 </head>
5 <body>
6   <h2>Document Object Model</h2>
7   
8   <p>
9     This is a simple
10    <code>HTML</code>
11    page to illustrate the
12    <a href="http://www.w3.org/DOM/">DOM</a>
13  </p>
14 </body>
15 </html>
```

OR

18. (a) Discuss how structured data extraction can be performed across multiple pages, such as following links and maintaining state during the extraction process. (8)
(b) Describe the concept of multiple alignment and how it can be applied to handle variations in web page structures during wrapper generation. (6)
19. (a) Explain the challenges and considerations in collecting and pre-processing web usage data, such as privacy concerns and data quality issues. (8)
(b) Identify the techniques used in query log mining, such as query clustering, query categorization, and query intent analysis. (6)

OR

20. (a) Make use of Web Usage Mining and explain how it can be applied to improve website recommendation systems and collaborative filtering. (8)
(b) Describe Click Stream Analysis and Web Server Log Files. (6)

B24CS3G11	INTRODUCTION TO DATA STRUCTURES	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course enables learners to understand fundamental data structures, their representations, and the operations performed on them. It introduces essential data structures such as arrays, stacks, queues, linked lists, trees, heaps, and graphs. Through this course, learners will develop the ability to select and apply the most appropriate data structures and related algorithms for solving computational and real-world problems efficiently.

Prerequisites

Knowledge in C programming

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Develop algorithms to solve problems and evaluate their performance based of time/space complexities expressed using asymptotic notations. (Cognitive Knowledge Level: Apply)
CO 2	Develop programs involving pointer operations, pointer-based function calls, dynamic memory handling, and function/structure pointers.(Cognitive Knowledge Level: Apply)
CO 3	Solve problems using simple data structures like arrays and queues for applications involving arithmetic expressions, sparse matrices, polynomials, searching etc.(Cognitive Knowledge Level: Apply)
CO 4	Construct dynamic data structures such as singly, doubly, and circular linked lists to efficiently manage memory and implement polynomial operations.(Cognitive Knowledge Level: Apply)
CO 5	Apply tree and graph data structures, searching and sorting techniques to represent and solve computing problems efficiently. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	3	2	3	3	3		2	1	3	2
CO2	2	2	2	2	2	1	2	2	2	2	2
CO3	2	3	2	3	3	2		2	2	3	2
CO4	2	3	2	3	3	2		2	2	2	2
CO5	3	3	2	3	3	1		1	1	1	1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Data Structures:

Types of Data Structures-Primitive, Non primitive-Linear, Nonlinear. Algorithm. Performance analysis-Time Complexity, Space Complexity, Complexity Calculation of Simple Algorithms

MODULE 2 (6 hours)

Pointers:

Pointers: declaration, operations on pointers, passing pointer to a function, pointer to pointer, array of pointers, pointer to function, pointer to structure.

MODULE 3 (8 hours)

Arrays:

Polynomial representation using Arrays, Sparse matrix, Stacks, Queues- Circular Queues, Priority Queues, Double Ended Queues, Evaluation of Expressions.

MODULE 4 (6 hours)

Linked List:

Self Referential Structures, Singly Linked List- Operations on Linked List, Circular Linked List, Doubly Linked List, Polynomial representation using Linked List.

MODULE 5 (10 hours)

Trees and Graphs:

Trees, Binary Trees- Tree Operations, Binary Tree Representation, Tree Traversals, Binary Search Trees- Binary Search Tree Operations. Graphs, Representation of Graphs, Depth First Search and Breadth First Search on Graphs. Searching-Binary search , Sorting-Bubble sort.

Text Books

1. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, Universities Press, Fundamentals of Data Structures in C.

Reference Books

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms, Fourth Edition, MIT Press.
2. Samanta D., Classic Data Structures, Prentice Hall India.
3. Richard F. Gilberg, Behrouz A. Forouzan, Data Structures: A Pseudocode Approach with C, 2/e, Cengage Learning.
4. Aho A.V., J.E. Hopcroft and J.D. Ullman, Data Structures and Algorithms, Pearson Publication.
5. Tremblay J.P. and P.G. Sorenson, Introduction to Data Structures with Applications, Tata McGraw Hill.
6. Wirth N., Algorithms + Data Structures = Programs, Prentice Hall.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Data Structures		6 hours
1.1	Types of Data Structures-Primitive, Non primitive-Linear, Nonlinear.	1
1.2	Algorithm	1
1.3	Performance analysis	1
1.4	Time Complexity,	1
1.5	Space Complexity,	1
1.6	Complexity Calculation of Simple Algorithms	1
Module 2: Pointers		6 hours
2.1	Pointers: declaration, operations on pointers	1

2.2	Passing pointer to a function	1
2.3	Pointer to pointer, Array of pointers	2
2.4	Pointer to function	1
2.5	Pointer to structure	1
Module 3: Arrays		8 hours
3.1	Polynomial representation using Arrays	1
3.2	Sparse matrix	1
3.3	Stacks	1
3.4	Queues	1
3.5	Circular Queues,	1
3.6	Priority Queues	1
3.7	Double Ended Queues,	1
3.8	Evaluation of Expressions	1
Module 4: Linked List		6 hours
4.1	Self Referential Structures	1
4.2	Singly Linked List	1
4.3	Operations on Linked List	1
4.4	Circular Linked List	1
4.5	Doubly Linked List	1
4.6	Polynomial representation using Linked List	1
Module 5: Trees and Graphs		10 hours
5.1	Trees, Binary Trees	1
5.2	Tree Operations	1
5.3	Binary Tree Representation	1
5.4	Tree Traversals	1
5.5	Binary Search Trees	1
5.6	Binary Search Tree Operations.	1
5.7	Graphs, Representation of Graphs	1
5.8	Depth First Search on Graphs	1
5.9	Breadth First Search on Graphs	1
5.10	Searching-Binary search , Sorting-Bubble sort	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Write an algorithm to search an element in a sorted array using Binary Search and determine its time complexity using Big-O notation.
2. Design an algorithm to find the maximum and minimum elements in an array with minimum comparisons, and analyze its time and space complexity.

Course Outcome 2 (CO2):

1. Write a C program to swap two numbers using pointers and pointer-based function calls. Explain how pointer parameters enable data modification.
2. Develop a C program that dynamically allocates memory for an array of integers using malloc(), reads values, computes the sum, and frees the allocated memory. Demonstrate the use of structure pointers in your solution.

Course Outcome 3 (CO3):

1. Using an array implementation of a stack, write a program to evaluate a postfix expression. Show how the stack changes at each step.
2. Represent a sparse matrix using a 3-tuple form and write a program to perform the fast transpose of the sparse matrix.

Course Outcome 4 (CO4):

1. Write a program to create a singly linked list and perform insertion and deletion at the beginning, end, and a given position.
2. Construct a polynomial using a linked list and write a program to add two polynomials represented using singly linked lists.

Course Outcome 5 (CO5):

1. Create a binary search tree (BST) by inserting a given set of integers and write functions to perform inorder, preorder, and postorder traversals.
2. Represent a graph using an adjacency list and implement BFS and DFS traversals for the graph.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CS3G11

Course Name: INTRODUCTION TO DATA STRUCTURES

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define asymptotic notations. Explain Big-O with an example.
2. Write the algorithm for Linear Search and analyze its time complexity.
3. Explain pointer to function with a suitable example.
4. Differentiate between pointer to pointer and array of pointers with examples.
5. Represent the following sparse matrix in tuple form.
6. Convert the infix expression $(A + B) * (C - D)$ into postfix form using stack rules.
7. List any three differences between singly, doubly, and circular linked lists.
8. Explain the steps to insert a node at the end of a singly linked list.
9. Write the preorder, inorder, and postorder traversals for a binary tree.
10. Differentiate between BFS and DFS traversal in graphs.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the classification of data structures with neat diagrams. Discuss primitive and non-primitive data structures, and differentiate between linear and non-linear data structures with suitable examples. (8)
- (b) Define an algorithm. Describe the essential characteristics of a good algorithm and write an algorithm for finding the largest element in an array. (6)

OR

12. (a) Write the algorithm for Binary Search and analyze its time complexity using Big-O, Ω and Θ . (8)
- (b) Compare iterative and recursive algorithms using an example. Analyze their time and space complexities. (6)
13. (a) Explain in detail the various operations that can be performed on pointers in C. Illustrate each operation (assignment, arithmetic, comparison, dereferencing) with suitable examples. (8)
- (b) Write a program to demonstrate passing a pointer to a function. Explain how call-by-address works with proper diagrams. (6)

OR

14. (a) Describe the concept of pointer to pointer, array of pointers, and pointer to function. Provide suitable code examples for each case. (8)
- (b) Define a structure in C and show how to create a pointer to a structure. Write a short program to access structure members using the arrow operator ($\rightarrow$) (6)
15. (a) Explain how polynomials are represented using arrays. Describe the method of storing coefficients and exponents, and write an algorithm to add two polynomials using array representation. (8)
- (b) What is a sparse matrix? Explain different types of sparse matrix representations and illustrate the tuple form with a suitable example. (6)

OR

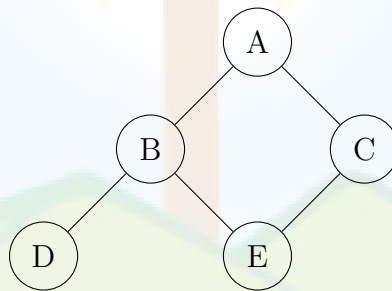
16. (a) Explain the array representation of a circular queue. Write the algorithm for insertion and deletion. (8)
- (b) Explain the evaluation of expressions using stacks. Demonstrate the evaluation of a postfix A B C * + D - or prefix expression - + A * B C D with step-by-step stack operations. (6)
17. (a) Compare singly, doubly, and circular linked lists in terms of structure, memory usage, and operations. (8)
- (b) Construct a singly linked list and write the algorithms for insertion at the beginning, at the end, and at a given position. (6)

OR

18. (a) Describe the structure and working of a circular linked list. Explain how insertion and deletion operations differ from a regular singly linked list with the help of diagrams. (8)
- (b) Explain how a polynomial can be represented using a linked list. Write the steps or algorithm for multiplying two polynomials using linked list representation. (6)
19. (a) What are tree traversals? Explain inorder, preorder, and postorder traversals with an example binary tree and show the step-by-step traversal sequences. (8)
- (b) Explain the representation of graphs using adjacency matrix and adjacency list. Compare their advantages. (6)

OR

20. (a) Explain binary search trees (BST). Discuss the operations of insertion, search, and deletion in a BST. (8)
- (b) Write the algorithm for Breadth First Search (BFS). Explain its working with an example graph. (6)



B24CS3G12	COMPUTER GRAPHICS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course provides a systematic foundation in the principles and computational techniques of computer graphics, emphasizing the mathematical models and algorithms used to represent and manipulate graphical data. Covers display technologies, drawing algorithms, 2D and 3D geometric transformations, clipping procedures, projections, and visible-surface detection methods. The course equips students with the analytical and technical skills required to design graphical components, develop rendering pipelines, and apply graphics concepts in advanced visualization and interactive systems.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Describe the working principles of graphics devices.(Cognitive Knowledge Level: Understand)
CO 2	Illustrate line drawing, circle drawing and polygon filling algorithms.(Cognitive Knowledge Level: Apply)
CO 3	Demonstrate geometric representations and transformations on 2D & 3D objects .(Cognitive Knowledge Level: Apply)
CO 4	Demonstrate the working of line and polygon clipping algorithms.(Cognitive Knowledge Level: Apply)
CO 5	Apply 3D viewing transformations, projection techniques, and visible surface detection algorithms for rendering 3D objects with awareness of responsible and ethical use in practical applications.(Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1		1					1		
CO 2	2	2	1	2	1				1		
CO 3	2	3	2	2	1				1		2
CO 4	2	3	2	3	3				1		2
CO 5	2	3	2	3	3	2	2		1		2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Basics of Computer Graphics:

Basics of Computer Graphics and its applications. Video Display devices- Refresh Cathode Ray Tubes(CRT), Random Scan Displays and systems, Raster scan displays and systems, Color CRT displays, Flat panel display and its categories.

MODULE 2 (8 hours)

Line drawing, Circle drawing and Filled Area Primitives:

Line drawing algorithms- DDA, Bresenham's algorithm. Circle drawing algorithms- Midpoint Circle generation algorithm, Bresenham's algorithm. Filled Area Primitives- Scan line polygon filling, Boundary filling and flood filling.

MODULE 3 (8 hours)

Geometric Transformations:

Two dimensional transformations-Translation, Rotation, Scaling, Reflection and Shearing, Composite transformations, Matrix representations and homogeneous coordinates. Basic 3D transformations

MODULE 4 (6 hours)

Clipping:

Window to viewport transformation. Cohen Sutherland and Midpoint subdivision line clipping algorithms, Sutherland Hodgeman and Weiler Atherton Polygon clipping algorithms.

MODULE 5 (8 hours)

3D Graphics:

Three dimensional viewing pipeline. Projections- Parallel and Perspective projections. Visible surface detection algorithms- Back face detection, Depth buffer algorithm, Scan line algorithm, A buffer algorithm

Text Books

1. Zhigang Xiang and Roy Plastock, Computer Graphics (Schaum's outline Series), McGraw Hill, 2019.
2. Donald Hearn and M. Pauline Baker, Computer Graphics, PHI, 2e, 1996

Reference Books

1. William M. Newman and Robert F. Sproull, Principles of Interactive Computer Graphics. McGraw Hill, 2001
2. David F. Rogers , Procedural Elements for Computer Graphics, Tata McGraw Hill, 2001.
3. Donald Hearn, M. Pauline Baker and Warren Carithers, Computer Graphics with OpenGL, PHI, 4e, 2013

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Basics of Computer Graphics		6 hours
1.1	Basics of Computer Graphics and applications	1
1.2	Refresh Cathode Ray Tubes	1
1.3	Random Scan Displays and systems	1
1.4	Raster scan displays and systems	1
1.5	Color CRT displays	1
1.6	Flat panel display and its categories.	1
1.7	Practice Problems	1
Module 2: Line drawing, Circle drawing and Filled Area Primitives		8 hours
2.1	DDA Line drawing Algorithm	1
2.2	Bresenham's line drawing algorithm	1
2.3	Midpoint Circle generation algorithm	1
2.4	Bresenham's Circle generation algorithm	1
2.5	Illustration of line drawing and circle drawing algorithms	1
2.6	Scan line polygon filling	2
2.7	Boundary filling and flood filling	1

Module 3:Geometric Transformation		8 hours
3.1	Basic 2D transformations-Translation and Rotation	1
3.2	Basic 2D transformations- Scaling	1
3.3	Reflection and Shearing	1
3.4	Illustration of 2D Transformations	1
3.5	Composite transformations	1
3.6	Matrix representations and homogeneous coordinates	1
3.7	Basic 3D transformations	1
3.8	Illustration of basic 3D transformations	1
Module 4: : 2D Clipping		6 hours
4.1	Window to viewport transformation	1
4.2	Cohen Sutherland Line clipping algorithm	1
4.3	Midpoint subdivision Line clipping algorithm	1
4.4	Sutherland Hodgeman Polygon clipping algorithm	1
4.5	Weiler Atherton Polygon clipping algorithm	1
4.6	Practice problems on Clipping algorithms	1
Module 5: : 3D graphics		8 hours
5.1	Three dimensional viewing pipeline, Projections-Parallel projections	1
5.2	Projections- Perspective projections	1
5.3	Visible surface detection algorithms- Back face detection.	1
5.4	Depth buffer algorithm	1
5.5	Depth buffer algorithm	1
5.6	Scan line visible surface detection algorithm	1
5.7	Scan line visible surface detection algorithm	1
5.8	A buffer algorithm	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Compare the working principle of raster scan systems and random scan systems.
2. How much time is spent scanning across each row of pixels during screen refresh on a raster system with resolution of 1280×1024 and a refresh rate of 60 frames per second?

Course Outcome 2 (CO 2):

1. A Rasterize the line using Bresenham's line drawing algorithm with end points (2,3) and (5,8) accepted from the user and implement it using any appropriate programming language. (Assignment)
2. Illustrate how the 4-connected boundary filling approach differs from 8-connected boundary filling and implement it using any appropriate programming language. (Assignment)

Course Outcome 3 (CO 3):

1. Rotate a triangle ABC 45 degree counter clockwise about the pivot point (10,3) , where the position vector of the coordinates is given as A(4,1), B(5,2) and C(4,3).
2. Implement the above transformation using any appropriate programming language with user inputs. (Assignment)
3. Illustrate the steps required for a general 3D rotation if the rotation axis is not parallel to any one of the principal axis. The rotation axis is defined by the points $P_1(x_1, y_1, z_1)$ and $P_2(x_2, y_2, z_2)$. Give its composite matrix representation.

Course Outcome 4 (CO 4):

1. Given a clipping window A(20,20), B(60,20), C(60,40) and D(20,40). Using Cohen Sutherland algorithm, find the visible portion of the line segment joining the points P(40,80) and Q(120,30).
2. Implement Cohen Sutherland clipping algorithm using any appropriate programming language with user inputs. (Assignment)

Course Outcome 5 (CO 5):

1. Apply histogram equalization to enhance a low-contrast grayscale image.
2. Explain scan line algorithm for detecting visible surfaces in an object.
3. Derive the matrix for performing perspective projection and parallel projection.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CS3G12

Course Name: COMPUTER GRAPHICS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Describe Flat Panel display and its categories.
2. Consider a raster system with a resolution of 1024*1024. Compute the size of the raster needed to store 4 bits per pixel? How much storage is needed if 8 bits per pixel are to be stored?
3. Justify the usage of integer arithmetic in Bresenham's line drawing algorithm.
4. How 8-way symmetry of circle can be used for developing circle drawing algorithms?
5. Show that two successive reflections about either of the coordinate axes is equivalent to a single rotation about the coordinate origin.
6. Determine a sequence of basic transformations that is equivalent to x-direction shearing.
7. Find the window to viewport normalization transformation with window lower left corner at (1,1) and upper right corner at (2,6).
8. How does Cohen Sutherland algorithm determine whether a line is visible, invisible or a candidate for clipping based on the region codes assigned to the end points of the line?

9. Define the terms (i) Centre of projection (ii) Principal vanishing point
10. Differentiate between the object space and image space method for the hidden surface removal of an image.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) . Explain the working principle of beam penetration method and shadow mask method with suitable illustrations. (8)
- (b) Draw the architecture of raster scan display systems and explain its working principle (6)

OR

12. (a) Explain the working principle of a Refresh CRT monitor with suitable diagrams. (8)
- (b) Describe random graphics system with suitable illustrations. (6)
13. (a) Differentiate between boundary fill and flood fill algorithms. (5)
- (b) Derive the initial decision parameter of Bresenham's line drawing algorithm and rasterize a line with endpoints (2,2) and (10,10). (9)

OR

14. (a) Write Midpoint circle drawing algorithm and identify the points in the circle with radius as 20 and center at (50,30) using the algorithm. (8)
- (b) Illustrate the working principle of scan line polygon filling algorithm. (6)
15. (a) Reflect a triangle ABC about the line $3x-4y+8=0$, where the coordinates of the triangle are given as A(4,1), B(5,2) and C(4,3). (8)
- (b) A diamond shaped polygon is located at P(-1,0), Q(0,-2), R(1,0) and S(0,2). Find the transformation matrix which would rotate the triangle by 90 degree counter clockwise about the point Q. Using the transformation matrix, find the coordinates of the rotated polygon. (6)

OR

16. (a) Describe the steps required for a general 3D rotation if the rotation axis is not parallel to any one of the principal axis. The rotation axis is defined by the points P1(x1,y1,z1) and P2(x2,y2,z2). Give its composite matrix representation. (8)
- (b) Consider a triangle at (2,2), (10,2), (2,10). Perform the following 2D transformations in succession and find the resultant vertices.
 - i. Scale with respect to (2,2) by scaling factors (2,2) along x and y directions.
 - ii. Rotate by 90 degree counter clockwise direction.

iii. Reflection based on $y=x$

(6)

17. (a) Illustrate Weiler – Atherton polygon clipping algorithm. (6)

(b) Explain Cohen-Sutherland line clipping algorithm. Use the algorithm to clip line with end points P1 (70, 20) and P2(100,10) against a window with lower left hand corner (50,10) and upper right hand corner (80,40). (8)

OR

18. (a) Describe Sutherland Hodgeman polygon clipping algorithm and list out its limitations. (7)

(b) Explain the steps involved in clipping a line using Mid point Subdivision algorithm. (7)

19. (a) Explain how visible surfaces can be detected using depth buffer algorithm (7)

(b) Define parallel projection. Describe orthographic and oblique parallel projection (7)

OR

20. (a) Illustrate the scan line method used in visible surface detection. (7)

(b) Derive the matrix needed for performing perspective projections. (7)

B24CS3G13	DIGITAL IMAGE PROCESSING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course helps the learners understand the core concepts and applications of Digital Image Processing. It covers digital image processing fundamentals; Image Transforms; Spatial and frequency domain filtering; Image segmentation; Morphological Image processing.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Understand the fundamental concepts and components of digital image processing, including image representation, sampling, and quantization. (Cognitive Knowledge Level: Understand)
CO 2	Apply various image transforms such as DFT, DCT, Walsh, and Hadamard to analyze image frequency characteristics. (Cognitive Knowledge Level: Apply)
CO 3	Implement image enhancement techniques in spatial and frequency domains using filtering and histogram-based methods. (Cognitive Knowledge Level: Apply)
CO 4	Compare and evaluate various image segmentation techniques including thresholding, region-based, and edge-based approaches in terms of their effectiveness for object detection. (Cognitive Knowledge Level: Apply)
CO 5	Utilize morphological operations and boundary representation techniques for image shape analysis and feature extraction. (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	1		1	1				1		1
CO 2	2	3	2	3	3			2	3	2	2
CO 3	3	3	3	3	3	3	1	3	3	3	3
CO 4	3	3	3	3	3	3	3	1	3	2	3
CO 5	2	3	2	3	3	3	2	3	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study (quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Fundamentals of Digital Image Processing

Introduction to Image processing and applications. Image as 2D data. Image representation in grayscale, Binary and Colour images. Fundamental steps in image processing. Components of image processing system. Pixels. Coordinate conventions; Imaging Geometry; Spatial Domain; Frequency Domain; sampling and quantization.

MODULE 2 (8 hours)

Image Transforms

Basic concept of spatial domain and frequency domain, Image transforms and its properties, Unitary transform; Discrete Fourier Transform; 2D DFT, 4 order DFT Transform coefficients, Discrete Cosine Transform; 2D DCT, 4 order DCT Transform Coefficients, Walsh Transform; Hadamard Transform.

MODULE 3 (8 hours)

Image Enhancement

Spatial domain methods : point processing- intensity transformations, image subtraction, image averaging, Histogram Processing-Equalization, Spatial filtering- smoothing filters, sharpening filters

Frequency domain methods: Smoothing Frequency Domain Filters: Ideal Low Pass Filter; Gaussian Low Pass Filter; Butterworth Low Pass Filter; Sharpening Frequency Domain Filters: Ideal High Pass Filter; Gaussian High Pass Filter; Butterworth High Pass Filter; Homomorphic Filtering.

MODULE 4 (7 hours)

Image Segmentation

Pixel-Based Approach- Multi-Level Thresholding, Local Thresholding, Threshold Detection Method; Region-Based Approach- Region Growing Based Segmentation, Region Splitting, Region Merging, Split and Merge, Edge Detection - Edge Operators; Line Detection, Corner Detection.

MODULE 5 (6 hours)

Morphological Operations

Basics of Set Theory; Dilation and Erosion - Dilation, Erosion; Structuring Element; Opening and Closing; Hit or Miss Transformation. Representation and Description Representation - Boundary, Chain codes, Polygonal approximation approaches, Boundary segments.

Text Books

1. A K. Jain, Fundamentals of digital image processing, Prentice Hall of India, 1989.
2. Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing (English) 3rd Edition, Pearson India, 2013

Reference Books

1. Al Bovik, The Essential Guide to Image Processing, Academic Press, 2009.
2. Milan Sonka, Vaclav Hlavac and Roger Boyle, Image Processing, Analysis, and Machine Vision, Thomson Learning, 2008 .
3. S Jayaraman, S Esakkirajan and T Veerakumar, Digital Image Processing, McGraw Hill Education, 2009 .

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Fundamentals of Digital Image Processing		7 hours
1.1	Introduction to Image processing and applications.	1
1.2	Image as 2D data. Image representation in grayscale, Binary and Colour images. Fundamental steps in image processing, Components of image processing system.	1
1.3	Pixels, basic relationship between pixels	1
1.4	Coordinate conventions, Imaging Geometry	1
1.5	Spatial Domain	1
1.6	Frequency Domain	1
1.7	Sampling and quantization.	1
Module -2 Image Transforms		8 hours
2.1	Basic concept of spatial domain and frequency domain.	1
2.2	Image transforms and its properties – Unitary transform	1
2.3	Discrete Fourier Transform, 2D DFT	1
2.4	4 order DFT Transform coefficients (No derivation)	1
2.5	Discrete Cosine Transform, 2D DCT	1

2.6	4 order DCT Transform Coefficients (No derivation)	1
2.7	Walsh Transform;	1
2.8	Hadamard Transform.	1
Module -3 Image Enhancement		8 hours
3.1	Spatial domain methods: point processing- intensity transformations, histogram processing	1
3.2	histogram processing	1
3.3	Spatial filtering- smoothing filters, sharpening filters	1
3.4	Smoothing frequency domain filters: ideal low pass filter; gaussian low pass filter;	1
3.5	Butterworth low pass filter	1
3.6	Frequency domain filters: ideal high pass filter; gaussian high pass filter	1
3.7	Butterworth high pass filter	1
3.8	Homomorphic filter	1
Module 4- Image Segmentation		7 hours
4.1	Pixel-Based Approach- Multi-Level Thresholding	1
4.2	Local Thresholding, Threshold Detection Method	1
4.3	Region-Based Approach- Region Growing Based Segmentation	1
4.4	Region Splitting, Region Merging, Split and Merge	1
4.5	Edge Detection - Edge Operators	1
4.6	Line Detection, Corner Detection	2
Module 5 - Morphological Operations		6 hours
5.1	Basics of Set Theory; Dilation and Erosion - Dilation, Erosion	1
5.2	Structuring Element; Opening and Closing	1
5.3	Hit or Miss Transformation	1
5.4	Representation and Description Representation	1
5.5	Boundary, Chain codes, Polygonal approximation approaches	1
5.6	Boundary segments	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Define grayscale image, binary image, and color image with suitable examples.
2. Explain the fundamental steps involved in a typical digital image processing system.
3. Analyze how image information changes when shifting from the spatial domain to the frequency domain using an image transform such as the DFT.
4. Apply the concept of sampling and quantization to convert a continuous image into a 2D digital image. Illustrate the process with a simple example or diagram.

Course Outcome 2 (CO 2):

1. Define the spatial domain and frequency domain in image processing. Give one example of an operation performed in each domain.
2. Compute the 4-point Discrete Fourier Transform (DFT) coefficients for the sequence $[1,0,1,0]$.
3. Analyze the differences between DFT, DCT, Walsh, and Hadamard transforms in terms of energy compaction, computational complexity, and typical applications.

Course Outcome 3 (CO 3):

1. List the different types of smoothing and sharpening filters used in spatial domain image processing.
2. Describe the process of histogram equalization and how it improves image contrast.
3. Analyze the differences between Ideal, Gaussian, and Butterworth Low Pass Filters in the frequency domain with respect to transition band, ringing, and practical applications.
4. Compare spatial domain and frequency domain filtering in terms of computational efficiency and edge preservation.
5. Describe the process of histogram equalization and how it improves image contrast.
6. Evaluate the effectiveness of Gaussian High Pass Filters for medical image enhancement compared to Ideal High Pass Filters.

Course Outcome 4 (CO 4):

1. List any two edge detection operators used for detecting image boundaries.

2. Describe the steps involved in region growing-based segmentation.
3. Apply multi-level thresholding to separate an image with three intensity groups (e.g., background, object, bright object) and specify suitable threshold values.
4. Analyze the differences between region splitting, region merging, and split-and-merge techniques. Discuss their computational impact.

Course Outcome 5 (CO 5):

1. Define dilation and erosion in morphological image processing.
2. Describe the concept of Hit-or-Miss transformation and its role in shape detection.
3. Compare and analyze polygonal approximation and chain code representation in terms of accuracy, compactness, and computational requirements.
4. Evaluate the suitability of boundary-based representation versus region-based representation for object recognition tasks.
5. Design a morphological processing sequence (using dilation, erosion, opening/closing) to remove noise and extract meaningful shapes from a binary image. Justify each step.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CS3G13

Course Name: DIGITAL IMAGE PROCESSING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. What are the fundamental steps in a basic digital image processing system?
2. Define sampling and quantization in the context of digital images. How do they affect image quality?
3. List any two properties of unitary transform.
4. State and prove the spatial shift and periodic property of 2D DFT.
5. Describe about contrast stretching in spatial domain.
6. Explain how sharpening can be done in the frequency domain using Gaussian high pass filter?
7. Explain the method of image segmentation using multilevel thresholding.
8. Explain different types of edge detection methods.
9. Differentiate between dilation and erosion.
10. Explain boundary segments.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Discuss spatial domain and frequency domain techniques. (7)
(b) Explain the complete process of converting a continuous image into a digital image through sensing, sampling, quantization, and encoding. (7)

OR

12. (a) Define a pixel. How is pixel depth related to image resolution? (7)
(b) Evaluate the advantages and limitations of different image acquisition methods. How does imaging geometry affect accuracy in applications? (7)
13. (a) Explain 4 properties of 2D Fourier Transform. (8)
(b) Define 1D and 2D Walsh Transformation Functions. (6)

OR

14. (a) Compare DFT and DCT in terms of energy compaction, computational complexity, and applications in image compression. (7)
(b) Explain the Hadamard Transform. Generate a 4×4 Hadamard matrix and discuss its orthogonality and sequency properties in image processing. (7)
15. (a) Explain the smoothing of images in frequency domain using (i) ideal low pass filters and (ii) Butterworth low pass filters. (7)
(b) Explain image subtraction and image averaging in spatial domain. Show mathematically how these operations help in image enhancement or noise reduction. (7)

OR

16. (a) What do you mean by histogram of an image? Explain about the histogram of basic image types. (7)
(b) Explain Butterworth filters for image smoothening and image sharpening. (7)
17. (a) Explain multi level thresholding and local thresholding for image segmentation. (7)
(b) Explain region splitting and merging. (7)

OR

18. (a) Define thresholding. Briefly explain about local thresholding. (7)
(b) How is a line detected? Give the masks for detecting a line. (7)
19. (a) Explain the following i) Polygonal approximation approaches ii) Boundary Segmentation (7)
(b) Explain and illustrate Hit or miss transform morphological algorithm with an example (7)

OR

20. (a) Define Chain Codes. (7)
(b) Describe in detail the morphological operations of dilation and erosion. Explain how they affect the shape and size of objects in a binary image with suitable examples. (7)

B24CS3L09	MICROPROCESSOR LAB	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	3	2		2024

Preamble

To provide practical exposure to the architecture, instruction set, and programming of the Intel 8086 microprocessor using MASM, focusing on arithmetic, logical, string, and data manipulation operations.

Prerequisites

Microprocessor systems

Course Outcomes

After the completion of the course the student will be able to

CO 1	Apply MASM and 8086 assembly programming tools to develop, assemble, link, and execute assembly programs. (Cognitive Knowledge Level: Apply)
CO 2	Apply 8086 instructions to implement arithmetic operations such as addition, multiplication, division, factorial, and integer square root. (Cognitive Knowledge Level: Apply)
CO 3	Apply assembly programming methods to perform array operations including searching, sorting, and summing selected elements. (Cognitive Knowledge Level: Apply)
CO 4	Apply string manipulation techniques in assembly to analyze, modify, and validate ASCII strings. (Cognitive Knowledge Level: Apply)
CO 5	Apply data handling techniques to accept keyboard input and implement numerical conversions and result display in assembly language programs. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	3	3	3	3	2	3	3	2	3
CO 2	2	3	2	3	3	2	1	2	2	2	3
CO 3	2	3	3	3	3	2	1	3	3	2	3
CO 4	2	3	3	3	3	2	1	3	3	2	3
CO 5	2	3	3	3	3	3	3	3	3	3	3

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Evaluation	20 marks
- Viva (Self Study)	10 marks
- Output	10 marks
Lab Record	5 marks
Viva-voce/Test	15 marks

SYLLABUS

LIST OF EXPERIMENTS

1	Familiarization of 8086 and study of MASM programs (assemble, link, run a sample .ASM).
2	Addition of two 8-bit numbers and display the result (with carry if any).
3	Addition of two 16-bit numbers and display the result (handle carry).
4	Multiplication of two 16-bit numbers and store 32-bit product in DX:AX.
5	Division of a 32-bit number (DX:AX) by a 16-bit divisor; display quotient and remainder.
6	Find the largest and smallest elements in an array.
7	Calculate the sum of even numbers in an array.
8	Sort an array of N numbers in ascending order.
9	Count uppercase and lowercase letters in a given ASCII string.
10	Check whether a given string is a palindrome.
11	Convert a hexadecimal number to its BCD equivalent.
12	Compute the factorial of a given number.
13	Find the integer square root of a number.

14	Concatenate two ASCII strings.
15	Find the diagonal sum of elements in a given matrix

NB:During the lab cycle, students must enter the required data using the keyboard and complete the assigned operations accordingly.

Reference Books

1. Ramesh S. Gaonkar, Microprocessor Architecture, Programming and Applications with the 8085/8086, Penram International.
2. K. Udaya Kumar and B. S. Uma, The 8085 Microprocessor: Architecture, Programming and Interfacing, Pearson.
3. A. K. Ray and K. M. Bhurchandi, Advanced Microprocessors and Peripherals, Tata McGraw-Hill.

B24CS3L10	MINI PROJECT	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	3	2	2	2024

Preamble

The Mini Project course is designed to provide students with practical exposure to the principles and practices of software development. It enables learners to apply core concepts of Software management system in identifying real-world problems and developing suitable technological solutions. Throughout the course, students follow the complete software development lifecycle, including literature survey, problem definition, requirement analysis, software design, implementation, testing, and deployment.

Prerequisites

Entrepreneurship and Software management Systems (B24HU2T03)

Course Outcomes

After the completion of the course the student will be able to

CO 1	Identify real-world problems that are technically viable, economically feasible, and address societal or environmental needs for software development.(Cognitive Knowledge Level: Apply)
CO 2	Conduct a comprehensive literature survey to understand existing solutions and gain exposure to relevant software development processes.(Cognitive Knowledge Level: Apply)
CO 3	Perform requirement analysis, select appropriate design methodologies, and develop adaptable, reusable, and low-complexity software solutions using modern tools and advanced programming techniques. (Cognitive Knowledge Level: Apply) .(Cognitive Knowledge Level: Apply)
CO 4	Prepare a well-structured technical report and deliver an effective presentation on the developed project, adhering to professional standards. (Cognitive Knowledge Level: Apply)
CO 5	Apply engineering and project management principles to plan, execute, and successfully complete the mini project while functioning effectively as a team member/leader and adhering to professional ethics.(Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	1		1		2	1			2	
CO 2		1		1					1		3
CO 3	3	3	3	2	3	2					
CO 4		1		1			1	1	3		
CO 5	1						2	3	1	2	



Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	50	50	

Continuous Internal Evaluation Pattern

Attendance	10 marks
Project Guide	15 marks
Project Report	5 marks
Evaluation by the Committee (will be evaluating the level of completion and demonstration of functionality/specifications, presentation, oral examination, work knowledge and involvement)	20 marks

End Semester Examination Pattern

Presentation	15 marks
Demo	15 marks
Viva	20 marks

Assessment Pattern

Student Groups with 2 or 4 members should identify a topic of interest in consultation with a Faculty/Advisor. Review the literature and gather information pertaining to the chosen topic. State the objectives and develop a methodology to achieve the objectives. Carryout the design/fabrication or develop codes/programs to achieve the objectives by strictly following steps specified in the teaching plan. Innovative design concepts, performance, scalability, reliability considerations, aesthetics/ergonomic, user experience and security aspects taken care of in the project shall be given due weight.

The progress of the mini project is evaluated based on a minimum of two reviews. The review committee may be constituted by the Head of the Department comprising HoD or a senior faculty member, Mini Project coordinator and project guide. The internal evaluation shall be made based on the progress/outcome of the project, reports and a viva-voce examination, conducted internally by a 3-member committee. A project report is required at the end of the semester. The project has to be demonstrated for its full design specifications.

TEACHING PLAN

Students are expected to follow the following steps.

1. Review of Literature and Identification of a problem.
2. Create an abstract with a problem statement, solution approach, technology stack, team, etc. and get department approval. Register Online course/ Collect study materials.
3. Create Software Requirements Specification (SRS Document).
4. Create Software Design Document (SDD). This may include designs like,
 - (a) System Architecture Design
 - (b) Application Architecture Design
 - (c) GUI Design (Mockups)
 - (d) API Design
 - (e) Database Design
 - (f) Technology Stack
5. Create Test Plan, Test Scenarios and Test Cases (Test Case Document) and Traceability Matrix
6. Create a Project Plan (with Modules, Tasks, Resources, Time schedule) [May use any project management tool or excel for this] – Choose and follow any software engineering concept.
7. Development
 - (a) Set coding standards
 - (b) Environment Setup
 - (c) Source Code Control Setup (Like Subversion(SVN), Git)
 - (d) Development
 - (e) Unit Testing
 - (f) Integration Testing
 - (g) Testing /Quality Assurance(QA)
 - i. Functional Testing
 - ii. Load Testing
 - iii. Report Bugs
8. Resolve Bugs and Retest
9. Deployment (of software from local development environment to a production environment)
10. Test Run and Get Results
11. Prepare Project Report

Guidelines for the Report preparation

A bonafide report on the mini project shall be submitted within one week after the final presentation. Minimum number of pages should be 40.

1. Prepare the project report in LaTeX according to IEEE standards.
2. Use Times New Roman font for the entire report – Chapter/Section Title – Times New Roman 18, Bold; Heading 2 – Times New Roman 16, Bold; Heading 3 – Times New Roman 14, Bold; Body- Times New Roman 12, Normal.
3. Line Spacing – Between Heading 2 – 3 lines, between lines in paragraph 1.5 lines.
4. Alignments – Chapter/Section Title – Center, Heading 2 and 3 should be Left Aligned. Ensure that all body text is paragraph justified.
5. Figures and Tables – Ensure that all Figures and Tables are suitably numbered and given proper names/headings. Write figure title under the figure and table title above the table.
6. Suggestive order of documentation
 - (a) Top Cover
 - (b) Title page
 - (c) Certification page
 - (d) Acknowledgement
 - (e) Abstract
 - (f) Table of Contents
 - (g) List of Figures and Tables
 - (h) Chapters
 - (i) Appendices, if any
 - (j) References/Bibliography

B24CSM61	WEB TECHNOLOGIES (Minor)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course provides a foundational understanding of core web technologies required to design and develop modern web applications. It introduces the evolution of the Internet and World Wide Web and covers client-side development using HTML5, CSS and JavaScript, server-side scripting using PHP, integration with MySQL for data-driven applications, JSON as a lightweight data interchange format, and a basic exposure to Laravel as a contemporary web application framework. On completion, students will be able to build structured, responsive and dynamic web pages and simple full-stack web applications.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Build well-structured static web pages using core HTML5 elements and semantic page structure.(Cognitive Knowledge Level: Apply)
CO 2	Apply CSS and JavaScript to style web pages, implement basic layouts and add interactive behaviour on the client side.(Cognitive Knowledge Level: Apply)
CO 3	Develop simple server-side programs using PHP constructs such as variables, control structures, arrays, functions and string operations. (Cognitive Knowledge Level: Apply)
CO 4	Connect PHP with MySQL and implement basic CRUD-based, form-driven, data-centric web applications using sessions and cookies.(Cognitive Knowledge Level: Apply)
CO 5	Use JSON for data interchange and explain the basic structure and request-response flow of web applications built with the Laravel framework.(Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2				1	1	2	2	1		3
CO 2	2	2	3	3	2	2	3	3	3	2	3
CO 3	2	3	3	3	3	3	1	2	3	3	2
CO 4	3	3	3	3	3	3	3	3	3	3	3
CO 5	2	3	3	3	3	3	3	3	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (8 hours)

Introduction to WWW and HTML5

Evolution of Internet and World Wide Web; basic web architecture, client-server model, HTTP overview, URL and MIME concepts. HTML5 document structure: `<!DOCTYPE html>`, head and body; semantic elements such as `<header>`, `<nav>`, `<main>`, `<section>`, `<article>`, `<aside>`, `<footer>`. Fundamental HTML elements: headings, paragraphs, line breaks, hyperlinks (internal and external), images with `alt`, ordered and unordered lists, basic tables. HTML forms: structure, common input elements, HTML5 input types (text, email, number, date, range, color, etc.), `datalist` and `autocomplete`. Meta tags, page title and viewport. Introduction to embedding audio and video using HTML5 elements.

MODULE 2 (8 hours)

Cascading Style Sheets (CSS)

Need for CSS; CSS syntax and rule structure. Selectors: element, class, id and basic combinators. Style inclusion methods: inline, internal and external style sheets; concept of specificity and conflict resolution. Box model: content, padding, border and margin. Properties for fonts, text, colors, backgrounds and borders. Basic layout using `div` and `span`, display modes. Introduction to responsive design: viewport meta tag, simple media queries and adapting layout to screen size.

MODULE 3 (8 hours)

JavaScript Essentials

Role of JavaScript in web pages; embedding JavaScript in HTML (internal and external scripts). Variables, data types, operators and expressions. Decision making with `if`, `if--else`, `switch`; loops using `for`, `while`, `do--while`. Functions: definition, parameters, return values and scope (basic idea). Arrays and simple array operations. Document Object Model (DOM) basics: accessing elements, reading and modifying content and attributes. Simple client-side form validation using JavaScript.

MODULE 4 (8 hours)

PHP Basics

Introduction to PHP; embedding PHP in HTML and basic execution model. PHP syntax, variables, constants, data types and operators. Flow control: conditional statements (if, if--else, switch) and loops (for, while, do--while, foreach). User-defined functions: definition, parameters, return values and basic scope rules. Arrays: indexed and associative arrays, basic operations and iteration. String handling functions and simple string processing. Handling HTML form data with PHP using superglobals such as \$_GET and \$_POST.

MODULE 5 (13 hours)

PHP–MySQL Integration, JSON and Laravel

Relational database basics using MySQL; tables, primary key and simple SQL statements. Connecting PHP with MySQL; executing SQL queries from PHP. Implementing basic CRUD operations (INSERT, SELECT, UPDATE, DELETE) using PHP and MySQL; displaying query results. State management using cookies and sessions. JSON syntax and data types; encoding PHP arrays and objects as JSON and decoding JSON data in PHP. Need for web application frameworks; overview of Laravel, major features and advantages. Basic Laravel application structure (folders, MVC concept); routing and controllers; views and templating idea; simple request–response flow and redirection (conceptual).

Text Books

1. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, *Internet & World Wide Web: How to Program*, 5th Edition, Pearson. [Module 1, 2, 3, 4]
2. Lindsay Bassett, *Introduction to JavaScript Object Notation: A To-the-Point Guide to JSON*, 1st Edition, O'Reilly. [Module 5]
3. Julie C. Meloni, *PHP, MySQL & JavaScript All in One, Sams Teach Yourself*, 5th Edition, Pearson. [Module 4]
4. Matt Stauffer, *Laravel Up and Running: A Framework for Building Modern PHP Apps*, 1st Edition, O'Reilly. [Module 5]

Reference Books

1. Robert W. Sebesta, *Programming the World Wide Web*, 8th Edition, Pearson Education Inc.
2. Larry Ullman, *PHP 6 and MySQL 5 for Dynamic Web Sites: Visual QuickPro Guide*, Pearson.
3. Eric van der Vlist, Danny Ayers, Erik Bruchez, Joe Fawcett, Alessandro Vernet, *Professional Web 2.0 Programming*, Wrox, Wiley-India edition.

4. DT Editorial Services, *Web Technologies Black Book 2018 (As per Mumbai University Syllabus): HTML, CSS3, JavaScript, jQuery, AJAX, PHP, XML, MVC and Laravel*, ISBN: 9789386052490.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction to WWW and HTML5		8 hours
1.1	Evolution of Internet and World Wide Web	1
1.2	Web architecture, client-server model and HTTP overview	1
1.3	URL and MIME concepts	1
1.4	HTML5 document structure: <code><!DOCTYPE html></code> , head and body	1
1.5	Semantic elements: <code><header></code> , <code><nav></code> , <code><main></code> , <code><section></code> etc.	1
1.6	Fundamental HTML elements: headings, paragraphs, line breaks	1
1.7	Hyperlinks, images, ordered and unordered lists	1
1.8	Tables, forms, HTML5 input types, meta tags, audio and video	1
Module 2: Cascading Style Sheets (CSS)		8 hours
2.1	Need for CSS, CSS syntax and structure of style rules	1
2.2	CSS selectors: element, class, id and basic combinators	1
2.3	Inline, internal and external style sheets; specificity and conflict resolution	1
2.4	Box model: content, padding, border and margin	1
2.5	Text, font and color properties	1
2.6	Background properties and borders	1
2.7	Basic layout using <code>div</code> , <code>span</code> and display modes	1
2.8	Introduction to responsive design: viewport and media queries	1

Module 3: JavaScript Essentials		8 hours
3.1	Role of JavaScript in web pages; embedding scripts in HTML	1
3.2	Variables, data types and operators in JavaScript	1
3.3	Expressions and basic input/output using dialogs	1
3.4	Decision making with if , if--else , switch	1
3.5	Loops: for , while , do--while	1
3.6	Functions: definition, parameters, return values and scope (basics)	1
3.7	Arrays and simple array operations	1
3.8	DOM basics and simple client-side form validation	1
Module 4: PHP Basics		8 hours
4.1	Introduction to PHP, embedding PHP in HTML, execution model	1
4.2	PHP syntax, variables, constants and data types	1
4.3	Operators and expressions in PHP	1
4.4	Conditional statements: if , if--else , switch	1
4.5	Loops: for , while , do--while , foreach	1
4.6	User-defined functions: definition, parameters, return values	1
4.7	Arrays: indexed and associative, basic operations and iteration	1
4.8	String handling and simple form data processing using \$ _GET and \$ _POST	1
Module 5: PHP–MySQL Integration, JSON and Laravel		13 hours
5.1	MySQL basics: databases, tables and primary keys	1
5.2	Basic SQL statements for table creation and data manipulation	1
5.3	Establishing a PHP–MySQL connection	1
5.4	Executing SELECT queries and displaying results in web pages	1
5.5	Implementing INSERT operations from PHP forms	1
5.6	Implementing UPDATE operations from PHP	1
5.7	Implementing DELETE operations from PHP	1

5.8	Cookies: setting, reading and deleting cookies in PHP	1
5.9	Sessions: creating, using and destroying sessions	1
5.10	JSON syntax and data types; encoding arrays/objects as JSON in PHP	1
5.11	Decoding JSON data in PHP and simple usage scenarios	1
5.12	Laravel overview: MVC concept and application structure	1
5.13	Basic routing, controllers and views in Laravel (conceptual)	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Construct a valid HTML document for your personal Profile registration page for a Job Site www.123Jobs.com. Add relevant HTML elements in a table, to accept a minimum of 10 different fields which includes your name, address, phone, email address, your picture, your college, your branch, fields for your personal history (minimum 3 fields), favourite theory and practical subjects (checkbox), Username, Password (password).
2. What is MIME? Give the expansion of MIME. List four examples for MIME types. State the reason why MIME type specification is necessary in a request-response transaction between a browser and server.
3. What is codec? Recognize the role of **controls** attribute in `<video>` & `<audio>` tag in HTML. Use the COVID vaccination promotional video `MySafety.mp4` in a web page with suitable HTML code, **autoplay** option enabled and displayed in a standard dimension 750×500.

Course Outcome 2 (CO 2):

1. Organize a sample web page for the event 'Raagam2021' at your campus and use embedded Style sheets to apply a minimum 5 styles. State the Style Specification format of embedded style sheets.
2. Write CSS style rules to implement the following in a web page:
 - (a) to display the content of hyperlinks with yellow background color and in italics
 - (b) to display the contents of unordered lists in bold and in Arial font
 - (c) to display a background image titled "birds.jpg" with no tiling.
3. Write the code for an HTML document with embedded JavaScript scripts, which initially displays a paragraph with text "Welcome" and a button titled "Click". When the button is clicked, the message "Hello from JavaScript" in bold should replace the paragraph text.

Course Outcome 3 (CO 3):

1. Write a PHP program to store the name and roll no of 10 students in an Associative Array and Use **foreach** loop to process the array and Perform **asort**, **rsort** and **ksort** in the array. Illustrate with suitable output data.
2. Design an HTML page which enters a given number, write a PHP program to display a message indicating, whether the number is odd or even, when clicking on the submit button.
3. Write a PHP program to compute the sum of the positive integers up to 100 using **do while**.

Course Outcome 4 (CO 4):

1. Write a PHP form handling program to verify the user authentication credentials of a web page using MySQL connection and store the userid value as a Session variable if the userid is valid.
2. Create a valid HTML document for yourself, including your name, address, and email address. Also add your college; your major and the course. Perform form handling in PHP and process the output using **POST** method.
3. Write an embedded PHP script which displays the factorial of all numbers from 1 to 10 in a table in the web page. The factorial should be calculated and returned from a function. The table headings should be "Number" and "Factorial".

Course Outcome 5 (CO 5):

1. What is Route Model Binding in Laravel? Which types of route model binding are supported in Laravel?
2. Explain how Laravel performs route handling using routes calling controller methods.
3. List the data types used in JSON. Explain the use of **parse()** and **stringify()** functions in JSON with examples.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CSM61

Course Name: WEB TECHNOLOGIES(Minor)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define World Wide Web (WWW). Distinguish between client-side and server-side web technologies with one example each. (3)
2. Write the general syntax of a URL. Rewrite the default URL of your university website <https://ktu.edu.in> by adding a subdomain named **research** and a web page named **faq.html**. Show the full URL. (3)
3. Explain the difference between block-level and inline elements in HTML with two examples for each type. (3)
4. What are CSS selectors? Explain element, class and id selectors with suitable examples. (3)
5. Define responsive web design. How do the viewport meta tag and media queries help in implementing responsive layouts? (3)
6. What is the Document Object Model (DOM)? Write a simple JavaScript code snippet that changes the background colour of the page when a button is clicked. (3)
7. How is form data submitted from a browser to a PHP script using the POST method? Write a short PHP code fragment to read and display a field named **username**. (3)
8. Differentiate between a cookie and a session in PHP. Mention one typical use-case for each. (3)

9. What is JSON? List any four JSON data types and write a sample JSON object representing a student with fields `name`, `rollNo` and `cgpa`. (3)
10. What is MVC architecture in the context of Laravel? Name the three components and briefly describe the role of each. (3)

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Design an HTML5 web page that displays a “Student Profile” registration form for a campus placement portal. The form should collect at least the following information: Name, Register Number, Branch, Email, Mobile Number, CGPA, Preferred Job Location (selection list with at least 3 options) and Upload Resume (file input). Use appropriate HTML5 input types wherever applicable. (8)
- (b) Explain the terms URL and MIME type. Give the general syntax of a URL and list any four commonly used MIME types with their meanings. (6)

OR

12. (a) Write the HTML code to embed a video file named `campus_tour.mp4` in a web page with the following requirements: the video should have controls, should autoplay, and be displayed with a width of 640 pixels and a height of 360 pixels. If the browser does not support HTML5 video, a suitable fallback message should be shown. (6)
- (b) Create an HTML5 document for a simple “Department Homepage” that uses semantic elements such as `<header>`, `<nav>`, `<main>`, `<section>` and `<footer>`. The page should contain a navigation menu with links to “About”, “Faculty” and “Contact” sections and at least one table or list. Write the complete HTML code. (8)
13. (a) Consider a basic HTML page with a navigation bar, main content area and footer. Write the CSS rules to: (i) create a horizontal navigation bar with links displayed side by side, (ii) apply a background colour to the navigation bar, and (iii) change the link text colour on hover. Assume suitable class or id names and show the relevant HTML and CSS code. (8)
- (b) Explain the CSS box model with the help of a neat diagram. Differentiate between margin and padding, and show with a short code example how changing margin and padding affects the layout. (6)

OR

14. (a) Write a CSS style sheet to implement the following for a web page: (i) set the body font to Arial with font size 14px, (ii) set the background image to `pattern.png` without repeating, (iii) set all `<h1>` headings to blue colour and centred, and (iv) set unordered list bullets to square style and list items to bold. Show the corresponding HTML snippet that uses these styles. (8)

- (b) What is the role of media queries in responsive web design? Write the CSS code for a simple media query that changes the layout of a two-column page to a single-column layout when the screen width is below 600px. (6)
15. (a) Write an HTML document with embedded JavaScript that reads two numbers from the user using input fields and, when a “Compute” button is clicked, displays their sum, difference, product and quotient in separate paragraphs on the same page. (8)
- (b) Explain with examples the use of `if`, `if--else` and `switch` statements in JavaScript. (6)

OR

16. (a) Using JavaScript and DOM manipulation, write a script that validates a simple login form with fields `username` and `password`. The script should check that neither field is left empty. If any field is empty, an appropriate error message should be displayed below the form without reloading the page. (8)
- (b) What is an array in JavaScript? Illustrate with code how to: (i) declare an array of five integers, (ii) traverse the array using a `for` loop and print each element, and (iii) use the `join()` method on the array. (6)
17. (a) Write a PHP script that reads an integer `n` from an HTML form and prints the factorials of all numbers from 1 to `n` in a table with two columns: “Number” and “Factorial”. Use a user-defined function to compute the factorial of a number. (8)
- (b) Explain with examples any four string handling functions in PHP. (6)

OR

18. (a) Write a PHP program to store the names and marks of 5 students in an associative array, where the key is the name and the value is the mark. Display the list of students in descending order of marks. (8)
- (b) Describe how input from an HTML form is retrieved in a PHP program using the `$_GET` and `$_POST` superglobals, with suitable examples. (6)
19. (a) Write a PHP form handling program for user registration with fields: Name, Email, Username and Password. When the form is submitted, the program should insert the data into a MySQL table named `users` after establishing the necessary database connection. Show the database table structure and PHP code. (8)
- (b) Describe the steps for establishing a PHP–MySQL connection using `mysqli` or `PDO`. Mention how errors can be handled during connection. (6)

OR

20. (a) Enumerate the data types supported in JSON. Write a JSON schema for a “Student” document with fields: `id`, `name`, `branch`, `semester` and `cgpa`. (8)
- (b) Explain briefly the following concepts in Laravel: (i) routing, (ii) controllers, and (iii) views. Show a simple example where a route calls a controller method which returns a view. (6)

B24CSM62	PYTHON FOR MACHINE LEARNING (Minor)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course introduces students to Python programming through hands-on practice in writing, testing, and debugging code. Learners will apply conditional, iterative, data-structure, and object-oriented concepts to develop efficient and reliable programs. The course also equips students with skills to process and visualize data using modules such as NumPy, Pandas, and Matplotlib.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Write, test, and debug Python programs using scripting, functions, and control structures for machine learning applications (Cognitive Knowledge level: Apply)
CO 2	Utilize Python data structures (Lists, Tuples, Sets, Dictionaries) and file handling to organize and preprocess datasets (Cognitive Knowledge level: Apply)
CO 3	Apply data processing techniques using NumPy, Pandas, and Matplotlib for data cleaning, manipulation, and visualization (Cognitive Knowledge level: Apply)
CO 4	Implement machine learning workflows including data preprocessing, feature engineering, model training, and evaluation using Scikit-learn (Cognitive Knowledge level: Apply)
CO 5	Develop end-to-end ML solutions applying supervised and unsupervised learning methods such as regression, classification, and clustering (Cognitive Knowledge level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	2	1	1	2	2	1	2	2		1
CO2	1	1	1	1	1	1	1	1	1	1	1
CO3	1	1	1	2	2	1	2	2	1		1
CO4	2	2	2	3	3	2	2	3	3	2	3
CO5	2	3	3	3	3	3	3	3	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (10 hours)

Python Basics for Machine Learning:

Python programming environment – Python scripting and interactive execution (IDLE, Jupyter Notebook). Software development process – Case Study. Python fundamentals – Data types, variables, strings, comments, input/output. Control flow – conditional statements and loops. Functions and modular programming. Working with text files. Introduction to virtual environments and package management (pip, conda).

MODULE 2 (8 hours)

Data Structures and Data Handling:

Lists – operations, slicing, list comprehension, searching, sorting. Tuples and sets – use cases in machine learning workflows. Dictionaries – operations, value access, nested dictionaries, reverse lookup. Working with dates and times. Case Study – Selecting suitable data structures for ML tasks.

MODULE 3 (9 hours)

Numerical and Data Processing with NumPy & Pandas:

Working with `os` and `sys`. File I/O – reading/writing text and binary files. NumPy – array creation, indexing, slicing, vectorized operations, broadcasting, matrix computations, random number generation. Data visualization – matplotlib basics, labels, ticks, legends, histograms, scatter plots. Working with CSV files. Pandas – Series, DataFrame creation, indexing, cleaning, merging, grouping, handling missing data.

MODULE 4 (8 hours)

Machine Learning Foundations:

Introduction to Machine Learning – supervised vs. unsupervised learning, features and labels, ML workflow. Data preprocessing – handling missing data, encoding categorical variables, feature scaling (normalization, standardization). Train-test split and evaluation strategies. Exploratory data analysis (EDA) for ML using Pandas and Matplotlib. Introduction to scikit-learn – datasets, transformers, estimators.

MODULE 5 (10 hours)

Machine Learning Algorithms and Model Evaluation:

Linear Regression – concepts, implementation, evaluation. Logistic Regression –

classification basics, decision boundaries. k-Nearest Neighbors (kNN) – distance metrics, classification examples. Decision Trees – splitting criteria, visualization. Model evaluation metrics: accuracy, precision, recall, F1-score, MSE, confusion matrix. Case Study – Build, train, test, and evaluate an end-to-end ML model using a real dataset. Introduction to model tuning – cross-validation, hyperparameter tuning (GridSearchCV).

Text Books:

1. Andreas C. Müller and Sarah Guido, *Introduction to Machine Learning with Python*, O'Reilly Media, 2016.
2. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 3/e, O'Reilly Media, 2023.
3. Wes McKinney, *Python for Data Analysis*, 2/e, O'Reilly Media, 2017.

Reference Books:

1. Sebastian Raschka and Vahid Mirjalili, *Python Machine Learning*, 3/e, Packt Publishing, 2019.
2. François Chollet, *Deep Learning with Python*, 2/e, Manning Publications, 2021.
3. Jake VanderPlas, *Python Data Science Handbook*, O'Reilly Media, 2016.
4. Allen B. Downey, *Think Python: How to Think Like a Computer Scientist*, 2/e, Green Tea Press, 2015.
5. <https://scikit-learn.org/stable/>

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Python Basics for Machine Learning		10 hours
1.1	Python installation, scripting, and interactive execution (IDLE, Jupyter Notebook).	1
1.2	Working with Python virtual environments (pip, conda).	1
1.3	Software development process — Case Study.	1

1.4	Python fundamentals: data types, variables, strings, comments, input/output.	1
1.5	Control flow: conditionals and loops.	1
1.6	Functions and modular programming.	1
1.7	Working with text files.	1
1.8	Error handling and debugging basics.	1
1.9	Using libraries and modules.	1
1.10	Hands-on examples and exercises.	1
Module 2: Data Structures and Data Handling		8 hours
2.1	Lists: operations, slicing, searching, sorting.	1
2.2	List comprehension and nested lists.	1
2.3	Working with tuples and sets.	1
2.4	Dictionaries: creation, access, update, reverse lookup.	1
2.5	Working with dates and times.	1
2.6	Data structure use cases in ML workflows.	1
2.7	File handling for structured data (JSON, text).	1
2.8	Case Study — Selecting suitable data structures for ML tasks.	1
Module 3: Numerical and Data Processing with NumPy & Pandas		9 hours
3.1	Introduction to <code>os</code> and <code>sys</code> .	1
3.2	File I/O — text and binary files.	1
3.3	NumPy: array creation, indexing, slicing.	1
3.4	Vectorized operations and broadcasting.	1
3.5	Matrix operations and random numbers.	1
3.6	Matplotlib — basic plots.	1
3.7	Matplotlib — labels, ticks, legends, visualizations.	1
3.8	Working with CSV files.	1
3.9	Pandas: reading, cleaning, merging, grouping, handling missing data.	1
Module 4: Machine Learning Foundations		8 hours
4.1	Introduction to Machine Learning — supervised vs. unsupervised learning.	1

4.2	ML workflow — data, features, labels.	1
4.3	Data preprocessing — missing data handling.	1
4.4	Encoding categorical variables.	1
4.5	Feature scaling — normalization and standardization.	1
4.6	Train–test split; evaluation strategies.	1
4.7	Exploratory data analysis (EDA) using Pandas and Matplotlib.	1
4.8	Introduction to scikit-learn — datasets, transformers, estimators.	1
Module 5: ML Algorithms and Model Evaluation		10 hours
5.1	Linear Regression — concepts and implementation.	1
5.2	Linear Regression — evaluation (MSE, RMSE).	1
5.3	Logistic Regression — binary classification.	1
5.4	Logistic Regression — decision boundaries, evaluation metrics.	1
5.5	k-Nearest Neighbors (kNN) — distance metrics and classification.	1
5.6	Decision Trees — structure, splitting criteria.	1
5.7	Decision Trees — visualization and interpretation.	1
5.8	Model evaluation — accuracy, precision, recall, F1-score, confusion matrix.	1
5.9	Case Study — build/training/testing a full ML model.	1
5.10	Introduction to model tuning — cross-validation, GridSearchCV.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the difference between script mode and interactive mode in Python. Write a simple Python script that loads a dataset and prints its basic statistics.
2. Write a Python function that takes a list of numbers as input and returns both the mean and standard deviation. Demonstrate the use of arguments and return values.

Course Outcome 2 (CO 2):

1. Given a list of student scores, write a Python program using lists, tuples, sets, and dictionaries to store the data and remove duplicate scores. Display the highest, lowest, and unique scores.
2. Write a Python program that reads a text file containing names and stores the frequency of each name in a dictionary. Print the top three most frequent names.

Course Outcome 3 (CO 3):

1. Using NumPy and Pandas, read a CSV file containing house price data (price, area, bedrooms). Clean the dataset by replacing missing values and normalizing the numeric columns.
2. Create a scatter plot of area vs. price using matplotlib and label the axes and title appropriately.

Course Outcome 4 (CO 4):

1. Train a linear regression model using scikit-learn to predict house prices. Write Python code to split the dataset, train the model, and print the model coefficients.
2. Modify the above task to compute the model's MAE (Mean Absolute Error) and RMSE (Root Mean Square Error).

Course Outcome 5 (CO 5):

1. Using a dataset containing customer information (age, income, spending score), implement K-Means clustering. Print the cluster centers and assign each customer to a cluster.
2. Extend the above by visualizing the clusters using matplotlib and labeling the cluster regions.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CSM62

Course Name: PYTHON FOR MACHINE LEARNING(Minor)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Explain the role of Python in Machine Learning. Discuss the advantages of using Python for data analysis and model development with suitable examples.
2. Write a Python program that reads a CSV file containing numerical data and computes the mean, median, and standard deviation of a selected column using functions.
3. Explain local and global scope in Python. Demonstrate with a code snippet involving a function used in a machine learning preprocessing step.
4. Discuss the different ways of formatting strings in Python (f-strings, format(),
5. Compare lists, tuples, dictionaries, and NumPy arrays in the context of machine learning data handling.
6. Discuss the following Pandas DataFrame methods with examples: head(), info(), describe(), drop(), groupby(), fillna().
7. What is overfitting in Machine Learning? Explain with an example using a simple Python model.
8. Explain exception handling in Python with try/except/else/finally. Provide an example involving reading a dataset file.

9. Write a short note on the `os` and `sys` modules in Python. Also explain `os.getcwd()` and `os.listdir()` with examples related to dataset management.
10. Describe the characteristics of CSV files and explain why they are commonly used in machine learning workflows.

PART B

Answer any one full question from each module. Each question carries 14 marks.

11. (a) Compare script mode and interactive mode in Python. Explain how they support rapid development in machine learning workflows. (8)
(b) Write a Python program to load a dataset using Pandas and print the shape, column names, and first five rows. (6)

OR

12. (a) Explain arithmetic, comparison, and logical operators with examples useful in data preprocessing. (8)
(b) Describe the different steps in the machine learning workflow with a neat diagram. (6)
13. (a) Write a Python code to remove missing values from a list of numbers and compute the average. (6)
(b) Given a list of temperature readings, write a Python program using dictionaries to count the occurrence of each reading and display the most frequent temperature. (8)

OR

14. (a) Write a Python program to normalize a numeric list using min-max normalization. (6)
(b) Write a Python function to check whether a given string (column value in dataset) is a palindrome using slicing only. (8)
15. (a) Write a Python function `column_frequency(df, column)` that prints values in non-increasing order of frequency using Pandas. (6)
(b) Write Python code to sort a NumPy array without using built-in sort functions. Use a user-defined sorting function. (8)

OR

16. (a) Illustrate the following NumPy array operations with examples: `reshape()`, `flatten()`, `concatenate()`, `unique()`, `transpose()`. (8)
(b) Write a Python program to validate a password using conditions (length, digit, uppercase, lowercase, special character). (8)

17. (a) Explain how to create and train a linear regression model using Scikit-learn. Write Python code to fit the model and display coefficients. (8)
- (b) Explain different types of machine learning: supervised, unsupervised, and reinforcement. Give one Python example for each. (6)

OR

18. (a) Write Python code to implement a Logistic Regression classifier for predicting whether a customer will buy a product. Show data splitting, training, and accuracy calculation. (8)
- (b) Write Python code to implement KNN (k-nearest neighbors) classification on a sample dataset. (6)
19. (a) Write Python code to perform matrix multiplication using NumPy and compute the transpose of the result. (6)
- (b) Given a CSV file of automobile data, write Python code using Pandas to: (1) clean missing values, (2) compute number of cars per company, (3) calculate average mileage, (4) display the highest priced car. (8)

OR

20. (a) Write Python code to create a CSV file containing names and ages of machine learning pioneers. (8)
- (b) Given a sales CSV file, write Python code to visualize: (i) monthly sales as a scatter plot, (ii) two product sales as a bar chart, (iii) percentage contribution of each product using a pie chart. (8)

B24CSM63	WIRELESS NETWORKS AND IOT APPLICATIONS (Minor)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course provides a structured theoretical foundation in wireless communication systems and their application in Internet of Things (IoT) environments. The course emphasises textbook-aligned concepts covering wireless fundamentals, wireless sensor networks, IoT architectures and protocols, home and personal area networks, and pervasive computing applications. The course equips students with the ability to analyse wireless and IoT systems from architectural, protocol, performance, and security perspectives.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Analyse wireless communication systems by applying propagation models, interference concepts, and multiple access techniques to evaluate link performance and capacity.(Cognitive Knowledge level: Understand)
CO 2	Design and evaluate wireless sensor network architectures by selecting appropriate MAC, routing, and energy-management protocols for resource-constrained environments.(Cognitive Knowledge level: Apply)
CO 3	Evaluate and compare Internet of Things architectures and application-layer protocols to select suitable communication models for different IoT deployment scenarios.(Cognitive Knowledge level: Apply)
CO 4	Assess home and personal area wireless networks by comparing IEEE 802.15.4, Zigbee, and Bluetooth-based technologies with respect to performance, scalability, and reliability.(Cognitive Knowledge level: Apply)
CO 5	Identify and justify security mechanisms and application models for pervasive and IoT systems while considering privacy, context awareness, and system constraints.(Cognitive Knowledge level: Apply)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	2		2	2				2		1
CO 2	3	2	2	2	2	2		1	1	2	1
CO 3	2	2	2	1	1	1		1	2	2	2
CO 4	2	2	2	2	1			1	2	2	1
CO 5	1	1	1	1	1	3	3		1		2

Mapping of Course Outcomes with Program Outcomes

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Wireless Communication Fundamentals:

Radio spectrum and bandwidth. Noise, signal-to-noise ratio (SNR), and Shannon capacity. Free space propagation and large-scale path loss models. Shadowing and small-scale fading. Delay spread and Doppler spread. Interference and frequency reuse. Multiple access techniques: FDMA, TDMA, CDMA, and OFDMA. Wireless performance metrics: bit error rate (BER), throughput, and channel capacity..

MODULE 2 (9 hours)

Wireless Sensor Networks:

Wireless sensor network architecture and design objectives. Sensor node components and deployment strategies. Medium access control protocols for sensor networks. Energy-efficient communication techniques. Routing protocols in wireless sensor networks. Time synchronization mechanisms. Data aggregation techniques. Power management and duty cycling..

MODULE 3 (9 hours)

Internet of Things Architecture and Protocols:

Introduction to Internet of Things and its characteristics. IoT reference architecture. Roles of devices, gateways, and cloud platforms. IoT communication models. Message Queuing Telemetry Transport (MQTT) architecture and operation. Constrained Application Protocol (CoAP). RESTful services for IoT. IoT data handling and messaging models..

MODULE 4 (9 hours)

Home and Personal Area Networks:

Home automation concepts and applications. IEEE 802.15.4 standard. Zigbee architecture and protocol stack. Bluetooth Classic and Bluetooth Low Energy (BLE). Network formation and device discovery. Addressing and routing mechanisms. Latency and reliability

considerations in home and personal area networks..

MODULE 5 (9 hours)

Pervasive Computing Applications and Security:

Pervasive computing concepts and characteristics. Mobile and embedded devices. Context-aware systems. Security challenges in pervasive and IoT systems. Authentication and access control mechanisms. Privacy issues. Smart environments. Applications of pervasive and IoT systems..

Text Books

1. T. S. Rappaport, Wireless Communications: Principles and Practice, Pearson.
2. H. Karl and A. Willig, Protocols and Architectures for Wireless Sensor Networks, Wiley
3. O. Hersent, D. Boswarthick, O. Elloumi, The Internet of Things: Key Applications and Protocols, Wiley.
4. C. Gomez and J. Paradells, Wireless Home Automation Networks, Artech House.
5. F. Adelstein et al., Fundamentals of Mobile and Pervasive Computing, McGraw-Hill.

Reference Books

1. A. S. Tanenbaum and D. J. Wetherall, Computer Networks, Pearson.
2. J. Zheng and A. Jamalipour, Wireless Sensor Networks, Wiley.
3. S. Misra et al., Introduction to Wireless Sensor Networks, Cambridge University Press.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Wireless Communication Fundamentals		9 hours
1.1	Radio spectrum and bandwidth	1
1.2	Noise, SNR and Shannon capacity	1
1.3	Propagation and path loss models	1

1.4	Shadowing and fading	1
1.5	Delay spread and Doppler spread	1
1.6	Interference and frequency reuse	1
1.7	Multiple access techniques	1
1.8	Wireless performance metrics	1
1.9	Review and problem discussion	1
Module 2: Wireless Sensor Networks		9 hours
2.1	WSN architecture and design goals	1
2.2	Sensor node components	1
2.3	MAC protocols for WSN	1
2.4	Energy-efficient communication	1
2.5	Routing protocols	1
2.6	Time synchronization	1
2.7	Data aggregation	1
2.8	Power management and duty cycling	1
2.9	Review and problem discussion	1
Module 3: IoT Architecture and Protocols		9 hours
3.1	Introduction to IoT	1
3.2	IoT reference architecture	1
3.3	Device, gateway and cloud roles	1
3.4	IoT communication models	1
3.5	MQTT architecture	1
3.6	CoAP fundamentals	1
3.7	RESTful IoT services	1
3.8	IoT data handling	1
3.9	Review and discussion	1
Module 4: Home and Personal Area Networks		9 hours
4.1	Home automation concepts	1
4.2	IEEE 802.15.4 standard	1
4.3	Zigbee architecture	1
4.4	Bluetooth Classic	1

4.5	Bluetooth Low Energy	1
4.6	Network formation	1
4.7	Device discovery	1
4.8	Latency and reliability	1
4.9	Review and discussion	1
Module 5: Pervasive Computing Applications and Security		9 hours
5.1	Pervasive computing concepts	1
5.2	Mobile and embedded devices	1
5.3	Context-aware systems	1
5.4	Security challenges	1
5.5	Authentication mechanisms	1
5.6	Access control	1
5.7	Privacy issues	1
5.8	Smart environments	1
5.9	Applications and review	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain free space propagation and path loss models.
2. Compare FDMA, TDMA, CDMA, and OFDMA.

Course Outcome 2 (CO 2):

1. Describe wireless sensor network architecture.
2. Explain routing protocols used in WSN.

Course Outcome 3 (CO 3):

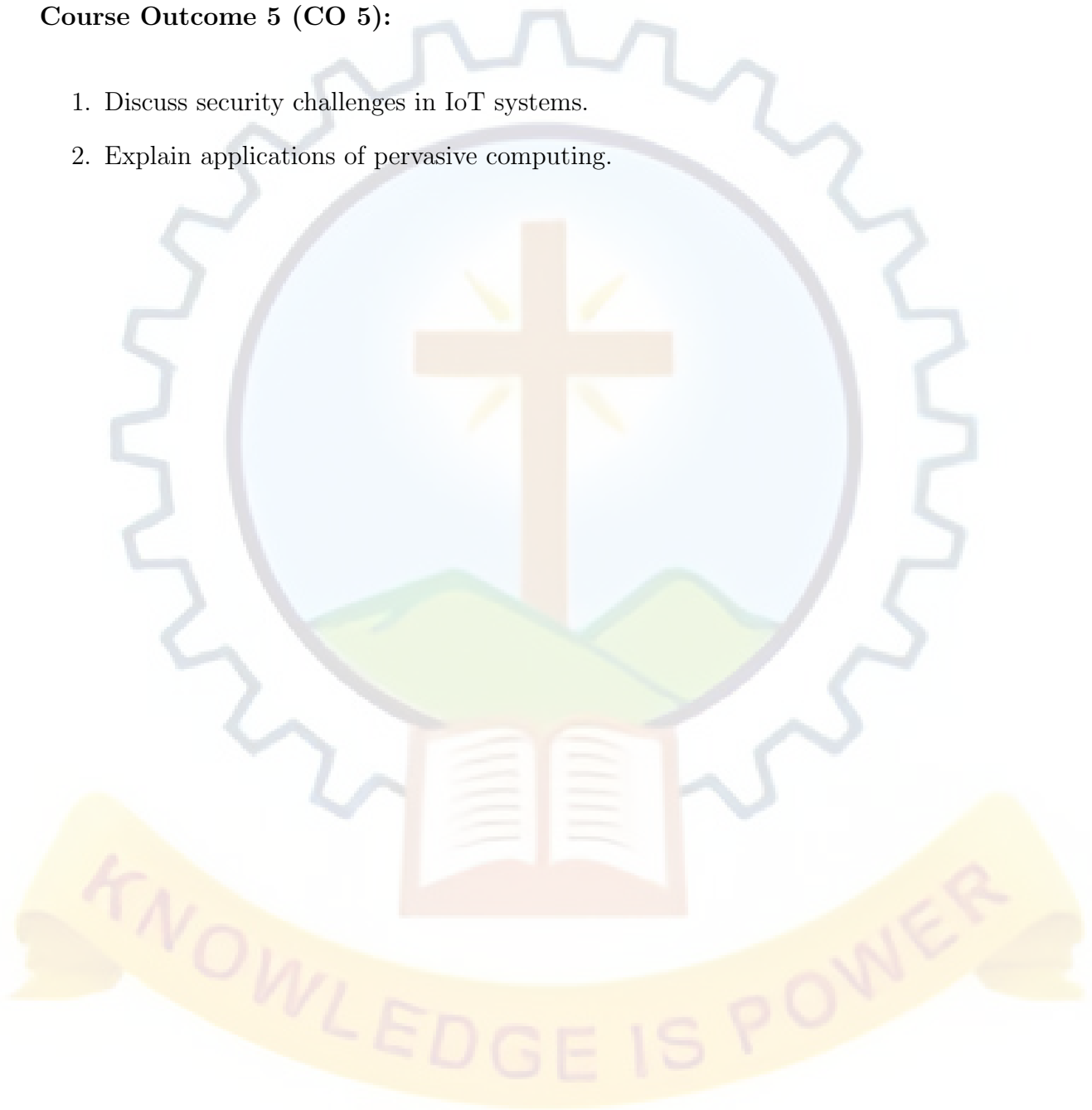
1. Explain IoT reference architecture.
2. Compare MQTT and CoAP protocols.

Course Outcome 4 (CO 4):

1. Explain Zigbee architecture for home automation.
2. Compare Bluetooth Classic and BLE.

Course Outcome 5 (CO 5):

1. Discuss security challenges in IoT systems.
2. Explain applications of pervasive computing.



MODEL QUESTION PAPER

QP CODE:

Pages: 4

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CSM63

***Course Name: WIRELESS NETWORKS AND IOT
APPLICATIONS(Minor)***

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define Shannon capacity. How is it related to signal-to-noise ratio?
2. Differentiate between shadowing and small-scale fading.
3. What is frequency reuse? State its significance in cellular systems.
4. List the major components of a wireless sensor node.
5. Explain the purpose of data aggregation in wireless sensor networks.
6. What are the roles of gateways in IoT architecture?
7. Compare MQTT and CoAP protocols.
8. Distinguish between Bluetooth Classic and Bluetooth Low Energy (BLE).
9. What is context awareness in pervasive computing?
10. List any three security challenges in IoT systems.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain radio spectrum, bandwidth, noise and SNR. Discuss Shannon capacity and its significance in wireless communication systems. (7)
- (b) Compare FDMA, TDMA, CDMA and OFDMA multiple access techniques with suitable illustrations. (7)

OR

12. Analyze the effects of large-scale path loss, shadowing, delay spread and Doppler spread on wireless communication performance. (14)
13. Explain the architecture of wireless sensor networks and discuss sensor node components, deployment strategies and design objectives. (14)

OR

14. (a) Describe medium access control protocols used in wireless sensor networks and evaluate their impact on energy consumption. (7)
- (b) Discuss routing protocols, time synchronization mechanisms and data aggregation techniques in wireless sensor networks. (7)
15. (a) Explain the IoT reference architecture and discuss the roles of devices, gateways and cloud platforms. (7)
- (b) Describe IoT communication models and data handling mechanisms with suitable examples. (7)

OR

16. A smart agriculture monitoring system uses distributed sensors and cloud-based analytics.
- (a) Explain how MQTT architecture supports efficient communication in such a system. (6)
- (b) Discuss the operation of CoAP and compare it with RESTful services for constrained IoT devices. (8)
17. (a) Explain the IEEE 802.15.4 standard and discuss the Zigbee architecture and protocol stack. (6)
- (b) Describe network formation, addressing and routing mechanisms in Zigbee networks. (8)

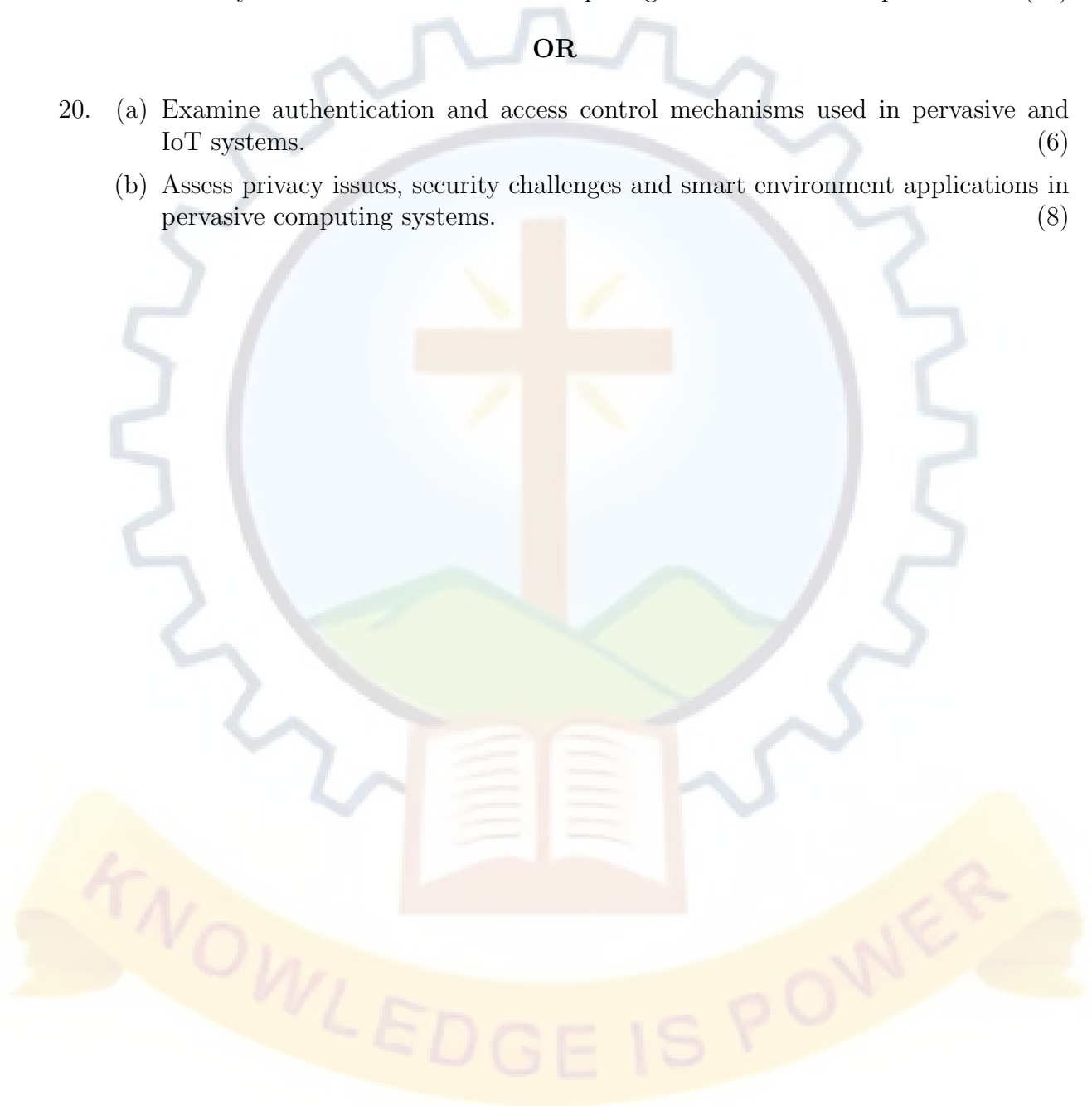
OR

18. A smart home environment consists of lighting systems, security sensors and wearable devices.

- (a) Compare Bluetooth Classic and BLE with respect to architecture, power consumption and applications. (6)
 - (b) Discuss latency and reliability considerations in home and personal area networks. (8)
19. Explain pervasive computing concepts and characteristics. Discuss mobile devices, embedded systems and context-aware computing with suitable examples. (14)

OR

20. (a) Examine authentication and access control mechanisms used in pervasive and IoT systems. (6)
- (b) Assess privacy issues, security challenges and smart environment applications in pervasive computing systems. (8)



B24CSH61	IMAGE AND VIDEO PROCESSING (Honours)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course aims to introduce the fundamentals of digital image and video representation, providing students with a strong foundation in visual information processing. It focuses on developing an understanding of various image enhancement, restoration, and transformation techniques essential for improving image quality and analysis. The course further explores segmentation, feature extraction, and motion estimation methods used in both static and dynamic visual data. In addition, it familiarizes students with image and video compression techniques and international coding standards. Finally, the course provides exposure to modern developments and emerging applications in image and video processing, including deep learning-based approaches.

Prerequisites

Problem solving and programming techniques (B24ES1T01A)

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Understand the fundamentals of image and video representation and perception. (Cognitive Knowledge Level: Understand)
CO 2	Apply image transformation and enhancement techniques for visual improvement.(Cognitive Knowledge Level: Apply)
CO 3	Perform image restoration and segmentation using spatial and frequency domain methods.(Cognitive Knowledge Level: Apply)
CO 4	Identify motion in videos and implement basic motion estimation techniques.(Cognitive Knowledge Level: Apply)
CO 5	Familiarise image and video compression methods and standards. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	1		1	1				1		1
CO 2	2	3	2	3	3	1		3	3	2	1
CO 3	2	3	3	3	3	2	3	3	3	2	1
CO 4	3	3	3	3	3	3	3	3	3	3	3
CO 5	3	3	3	3	3	3	3	3	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (8 hours)

Fundamentals of Image and Video Processing:

Digital image formation: image acquisition, sampling and quantization, Human visual perception and color vision models, Image representation: pixel relationships, neighborhood operations, Color spaces: RGB, HSV, YCbCr, HSI, Multi-dimensional signal processing concepts – 2D/3D signals. Basics of video signals – analog and digital video.

MODULE 2 (8 hours)

Image Transforms and Representations:

2D Discrete Transforms: DFT, DCT, and Wavelet transforms. Image pyramids and multi-resolution representations (Gaussian and Laplacian). Image registration and geometric transformations (translation, rotation, scaling). Image stitching and mosaicing.

MODULE 3 (10 hours)

Image Enhancement and Restoration

Spatial domain enhancement: contrast stretching, histogram equalization, smoothing and sharpening filters. Frequency domain enhancement: low-pass, high-pass, band-pass filtering. Image degradation and restoration models – inverse and Wiener filtering. Morphological image processing: erosion, dilation, opening, closing, skeletonization, boundary extraction.

MODULE 4 (9 hours)

Image Segmentation and Feature Extraction

Edge detection operators: Sobel, Prewitt, Laplacian, Canny. Region-based segmentation: thresholding, region growing, region splitting and merging, watershed. Feature extraction and description: corners, blobs, texture features, SIFT, SURF, Motion features for video sequences.

MODULE 5 (10 hours)

Video Processing and Compression

Video motion analysis: optical flow, block matching, motion estimation and compensation. Video stabilization and panoramic stitching. Fundamentals of image and video compression: redundancy reduction, lossless and lossy compression. Image compression standards: JPEG, JPEG2000. Video compression standards: MPEG-1/2/4, H.264, H.265. Introduction to modern techniques: deep learning-based image and video processing, CNNs, autoencoders, GANs.

Text Books

1. Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, Pearson Education.
2. Yao Wang, Joern Ostermann, Ya-Qin Zhang, Video Processing and Communications, Pearson Education.

Reference Books

1. Alan Bovik, Handbook of Image and Video Processing, Academic Press.
2. Anil K. Jain, Fundamentals of Digital Image Processing, Prentice Hall.
3. Milan Sonka, Vaclav Hlavac, Roger Boyle, Image Processing, Analysis and Machine Vision, Thomson Learning

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Digital image formation		8 hours
1.1	Introduction to Digital Image Formation and Acquisition	1
1.2	Sampling and Quantization	1
1.3	Human visual perception and color vision models	1
1.4	Image representation: pixel relationships	1
1.5	neighborhood operations	1
1.6	Color spaces: RGB, HSV, YCbCr, HSI	1
1.7	Multi-dimensional signal processing concepts – 2D/3D signals	1
1.8	Basics of video signals – analog and digital video.	1
Module 2: Image Transforms and Representations		8 hours
2.1	Introduction to 2D Discrete Transforms: DFT	1
2.2	DCT and Wavelet transforms	1

2.3	Image pyramids	1
2.4	Multi-resolution representations (Gaussian and Laplacian)	1
2.5	Image registration	1
2.6	geometric transformations (translation, rotation, scaling	1
2.7	Image stitching	1
2.8	Image mosaicing	1
Module 3: Image Enhancement and Restoration		10 hours
3.1	Introduction to Image Enhancement	1
3.2	Contrast Stretching and Intensity Transformations	1
3.3	Histogram Processing	1
3.4	Spatial Domain Filtering (Smoothing)	1
3.5	Spatial Domain Filtering (Sharpening)	1
3.6	Frequency Domain Enhancement Basics	1
3.7	Low-Pass, High-Pass and Band-Pass Filtering	1
3.8	Image Degradation Model	1
3.9	Restoration Techniques: Inverse and Wiener filtering.	1
3.10	Morphological Image Processing - erosion, dilation, opening, closing, skeletonization, boundary extraction.	1
Module 4: Image Segmentation and Feature Extraction		9 hours
4.1	Introduction to Segmentation	1
4.2	Edge Detection fundamenals, Edge detectors : Sobel and Prewitt	1
4.3	Laplacian and Canny Edge Detector	1
4.4	Thresholding-based Segmentation	1
4.5	Region-based Segmentation : region growing, region splitting and merging, watershed segmentation	1
4.6	Feature Extraction Basics : Corners,blobs	1
4.7	Texture Features	1
4.8	Advanced Feature Descriptors: SIFT, SURF	1
4.9	Motion Features for Video Sequences	1
Module 5: Video Processing and Compression		10 hours
5.1	Introduction to Video Processing	1

5.2	Optical Flow Basics	1
5.3	Block Matching , Motion Estimation	1
5.4	Motion Compensation and Applications Topics	1
5.5	Video Stabilization	1
5.6	Panoramic Stitching	1
5.7	Image and Video Compression Basics: redundancy reduction, lossless and lossy compression.	1
5.8	Image Compression Standards: JPEG, JPEG2000	1
5.9	Video compression standards: MPEG-1/2/4, H.264, H.265.	1
5.10	Deep Learning Techniques : CNNs, autoencoders, GANs	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain spatial sampling in image digitization. Why is proper sampling necessary?
2. Describe quantization with examples. Explain how bit depth affects image quality.
3. Describe the complete process of converting a real-world scene into a digital image, highlighting the role of sampling, quantization and color spaces.

Course Outcome 2 (CO 2):

1. Construct two levels of a Gaussian pyramid for the image using 2×2 averaging.

$$\begin{bmatrix} 10 & 14 & 18 & 22 \\ 12 & 16 & 20 & 24 \\ 14 & 18 & 22 & 26 \\ 16 & 20 & 24 & 28 \end{bmatrix}$$

2. Construct the corresponding Laplacian pyramid for level 1.
3. Two partially overlapping images of a campus are given. Apply a complete image stitching procedure that includes:

- (a) detecting feature points,
- (b) estimating geometric transformation (homography),
- (c) aligning images, and
- (d) blending using a Laplacian pyramid.

Describe and illustrate each step with example input/output.

Course Outcome 3 (CO 3):

1. Apply erosion and dilation using the structuring element

$$S = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

to a given 6×6 binary image.

2. Perform opening, closing, skeletonization, and boundary extraction for a simple binary shape.

Course Outcome 4 (CO 4):

1. Perform region growing from seed pixel (x=3, y=3) with threshold $T = 10$ on the given intensity set.
2. Explain the steps involved in Canny edge detection.
3. Describe the difference between region growing, region splitting, and region merging.

Course Outcome 5 (CO 5):

1. Explain the concept of optical flow and describe the difference between global and local optical flow methods (Horn–Schunck vs Lucas–Kanade).

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CSH61

Course Name: IMAGE AND VIDEO PROCESSING (Honours)

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Differentiate sampling and quantization in image digitization.
2. Explain luminance and chrominance in YCbCr color space.
3. Compute the 2- point DFT of the sequence $\{1, -1\}$.
4. Apply translation $T(-2, 3)$ to the point $(6, 4)$.
5. Apply contrast stretching to map the intensity range $[50, 150]$ of an image to $[0, 255]$.
6. Define image degradation and give two examples of degradation sources.
7. Explain the difference between Sobel and Prewitt edge detection operators.
8. Apply Otsu's method to the following histogram values and identify the optimal threshold: Grey levels = $\{0, 1\}$ with counts $\{30, 70\}$
9. What is block matching in motion estimation?
10. Write two differences between lossless and lossy compression.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain pixel adjacency concepts: 4-connected, 8-connected, and m-connected with diagrams. (7)
- (b) Describe neighborhood operations using masks. Give examples of 3×3 smoothing and sharpening masks. (7)

OR

12. (a) Explain the RGB, HSV, and HSI color models with diagrams. (7)
- (b) Describe the YCbCr color model and explain why it is used in digital video applications. (7)
13. (a) Apply translation $T(5, -3)$ and rotation 90° to point $(2, 3)$. Show the composite transformation matrix. (7)
- (b) Apply scaling $S(2, 0.5)$ to the square with vertices $(0,0)$, $(1,0)$, $(1,1)$, $(0,1)$. Give new vertex coordinates. (7)

OR

14. (a) Construct two levels of a Gaussian pyramid for the image using 2×2 averaging. (7)

$$\begin{bmatrix} 10 & 14 & 18 & 22 \\ 12 & 16 & 20 & 24 \\ 14 & 18 & 22 & 26 \\ 16 & 20 & 24 & 28 \end{bmatrix}$$

(7)

- (b) Construct the corresponding Laplacian pyramid for level 1. (7)
15. (a) Explain the need for smoothing filters and describe mean, median, and Gaussian smoothing. (7)
- (b) Discuss sharpening using Laplacian and unsharp masking with examples. (7)

OR

16. (a) Apply erosion and dilation using the structuring element

$$S = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

to a given 6×6 binary image.

- (b) Perform opening, closing, skeletonization, and boundary extraction for a simple binary shape.
- 17. (a) Explain the steps involved in Canny edge detection. (7)
- (b) Compare Sobel, Prewitt, and Laplacian operators. (7)

OR

- 18. (a) Explain the concept of corners and how Harris corner detection identifies them. (7)
- (b) Describe how blob features differ from corner features. (7)
- 19. (a) Explain the concept of optical flow and describe the difference between global and local optical flow methods (Horn–Schunck vs Lucas–Kanade). (7)
- (b) Describe the major differences between JPEG, JPEG2000, H.264, and H.265 standards. Focus on the types of redundancy removed and core algorithms used. (7)

OR

- 20. (a) A video contains camera shake. Describe how you would apply motion estimation for video stabilization (7)
- (b) Apply JPEG steps (DCT → Quantization → Zigzag → RLE) to compress a small 8×8 block. (7)

B24CSH62	COMPUTATIONAL HEALTH INFORMATICS (Honours)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		2024

Preamble

This course helps learners to develop know-how in computational methods, algorithms, and tools commonly used in health informatics. This includes data mining, machine learning, statistical analysis, and visualization techniques. Also, the course helps to gain knowledge of applications of machine learning in healthcare and how to analyze medical images, interpret healthcare data, and understand the role of informatics in disease diagnosis

Prerequisites

Basic background in Programming, Computational Biology and Machine learning

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Describe health informatics, including its principles, concepts, and applications of computational methods and techniques used in health informatics (Cognitive knowledge level: Understand)
CO 2	Illustrate latest trends, advancements, and emerging technologies in computational health informatics (Cognitive knowledge level: Apply)
CO 3	Demonstrate application of computational methods and techniques to analyze and manipulate medical images for various purposes, such as diagnosis, treatment planning, and research (Cognitive knowledge level: Apply)
CO 4	Use the machine learning techniques to health images to aid in various aspects of healthcare, including diagnosis, treatment planning, and disease monitoring (Cognitive knowledge level: Apply)
CO 5	Apply deep learning techniques to analyze and interpret medical images (Cognitive knowledge level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	2	1	1		1	2	2	2	1	1	1
CO2	1	1	1	1	1	1	1	2		1	2
CO3	3	2	1	2	2	1		2	1	1	1
CO4	2	3	3	3	3	3	3	3	2	2	2
CO5	2	3	3	3	3	3	3	3	2	2	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Introduction to Health Informatics

Definition, scope, and objectives of health informatics. Historical development and current trends. Health informatics frameworks and models. Health data standards (HL7, SNOMED CT, ICD). Interoperability challenges and solutions. Data capture, storage, and retrieval. Data quality and integrity. Data analytics techniques in healthcare. Data visualization for decision support

MODULE 2 (9 hours)

Emerging Technologies in Health Informatics

Artificial intelligence and machine learning in healthcare. Internet of Things (IoT) and healthcare applications. Hybrid IoT-NG-PON system. Blockchain technology in health informatics. Clinical research informatics. Genome sequencing and translational bioinformatics. IoT devices, architecture, and beneficiaries. Data sharing and secondary use of health data.

MODULE 3 (10 hours)

Medical Image Processing

Overview and significance of medical image processing. Challenges and opportunities. Principles of X-ray imaging. MRI basics. CT fundamentals. Ultrasound imaging characteristics. Image enhancement techniques. Contrast enhancement methods. Noise reduction and denoising. Image sharpening and edge enhancement.

MODULE 4 (8 hours)

Machine Learning in Medical Image Analysis

Image segmentation. Thresholding techniques. Region-based segmentation algorithms. Edge detection and contour-based segmentation. Feature extraction and representation. Supervised and unsupervised learning algorithms. Classification and regression techniques. Performance evaluation and validation of ML models

MODULE 5 (9 hours)

Deep Learning for Medical Image Processing

Convolutional Neural Networks (CNNs) for medical image analysis. Segmentation and object detection using deep learning. Transfer learning and pretrained models. Volumetric image analysis and 3D reconstruction. Image-based modeling and simulation. Advanced imaging modalities (functional MRI, diffusion tensor imaging). Artificial intelligence in medical image processing

Text Books

1. Translational Bioinformatics in Healthcare and Medicine. (2021). Netherlands: Elsevier Science.
2. Computational Analysis and Deep Learning for Medical Care: Principles, Methods, and Applications. (2021). United Kingdom: Wiley.

Reference Books

1. Introduction to Computational Health Informatics. United States (2020) CRC Press.
2. Signal Processing Techniques for Computational Health Informatics. (2020). Germany: Springer International Publishing.
3. Computational Intelligence Informatics. (2021). United Kingdom: Wiley.
4. Computational Intelligence for Machine Informatics. (2020). Germany: De Gruyter.
5. Smart Computational Intelligence and Learning in Biomedical Healthcare and Healthcare Health Informatics. (2021). United States: CRC Press.
6. Healthcare Systems and Health Informatics: Using Internet of Things.(2022). United States: CRC Press.
7. Deep Learning Techniques for Biomedical and Health Informatics. (2020).United Kingdom: Elsevier Science.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction to Health Informatics		9 hours
1.1	Definition, scope, and objectives of health informatics	1
1.2	Historical development and current trends	1
1.3	Health informatics frameworks and models	1
1.4	Health data standards (HL7, SNOMED CT, ICD)	1
1.5	Interoperability challenges and solutions	1
1.6	Data capture, storage, and retrieval	1

1.7	Data quality and integrity	1
1.8	Data analytics techniques in healthcare	1
1.9	Data visualization for decision support	1
Module 2: Emerging Technologies in Health Informatics		9 hours
2.1	Artificial intelligence and machine learning in healthcare	1
2.2	Internet of Things (IoT) and healthcare applications	1
2.3	Hybrid IoT-NG-PON system	1
2.4	Blockchain technology in health informatics	1
2.5	Clinical research informatics	1
2.6	Genome sequencing and translational bioinformatics	1
2.7	IoT devices and architecture	1
2.8	IoT beneficiaries in healthcare	1
2.9	Data sharing and secondary use of health data	1
Module 3: Medical Image Processing		10 hours
3.1	Overview and significance of medical image processing	1
3.2	Challenges and opportunities	1
3.3	Principles of X-ray imaging	1
3.4	MRI basics	1
3.5	CT fundamentals	1
3.6	Ultrasound imaging characteristics	1
3.7	Image enhancement techniques	1
3.8	Contrast enhancement methods	1
3.9	Noise reduction and denoising	1
3.10	Image sharpening and edge enhancement	1
Module 4: Machine Learning in Medical Image Analysis		8 hours
4.1	Image segmentation	1
4.2	Thresholding techniques	1
4.3	Region-based segmentation algorithms	1
4.4	Edge detection and contour-based segmentation	1
4.5	Feature extraction and representation	1
4.6	Supervised learning algorithms	1

4.7	Unsupervised learning algorithms	1
4.8	Performance evaluation and validation of ML models	1
Module 5: Deep Learning for Medical Image Processing		9 hours
5.1	Convolutional Neural Networks (CNNs) for medical image analysis	1
5.2	Segmentation and object detection using deep learning	1
5.3	Transfer learning and pretrained models	1
5.4	Volumetric image analysis and 3D reconstruction	1
5.5	Image-based modeling and simulation	1
5.6	Functional MRI	1
5.7	Diffusion tensor imaging	1
5.8	Artificial intelligence in medical image processing	1
5.9	Advanced imaging modalities (additional AI-driven techniques)	1

COURSE LEVEL ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Summarize Health informatics frameworks and models.
2. Explain Health data standards HL7, SNOMED CT and ICD.
3. Illustrate data analytics techniques and applications in healthcare.

Course Outcome 2 (CO2):

1. Explain Blockchain technology in health informatics.
2. Illustrate Internet of Things (IoT) and its applications in healthcare with examples.
3. How can translational bioinformatics facilitate the identification of disease-associated genetic variants and the development of targeted therapies?

Course Outcome 3 (CO3):

1. Differentiate principles and fundamentals of X-ray imaging, MRI, and CT.
2. Explain the importance of image enhancement techniques in medical imaging and how they contribute to improved diagnosis and treatment.

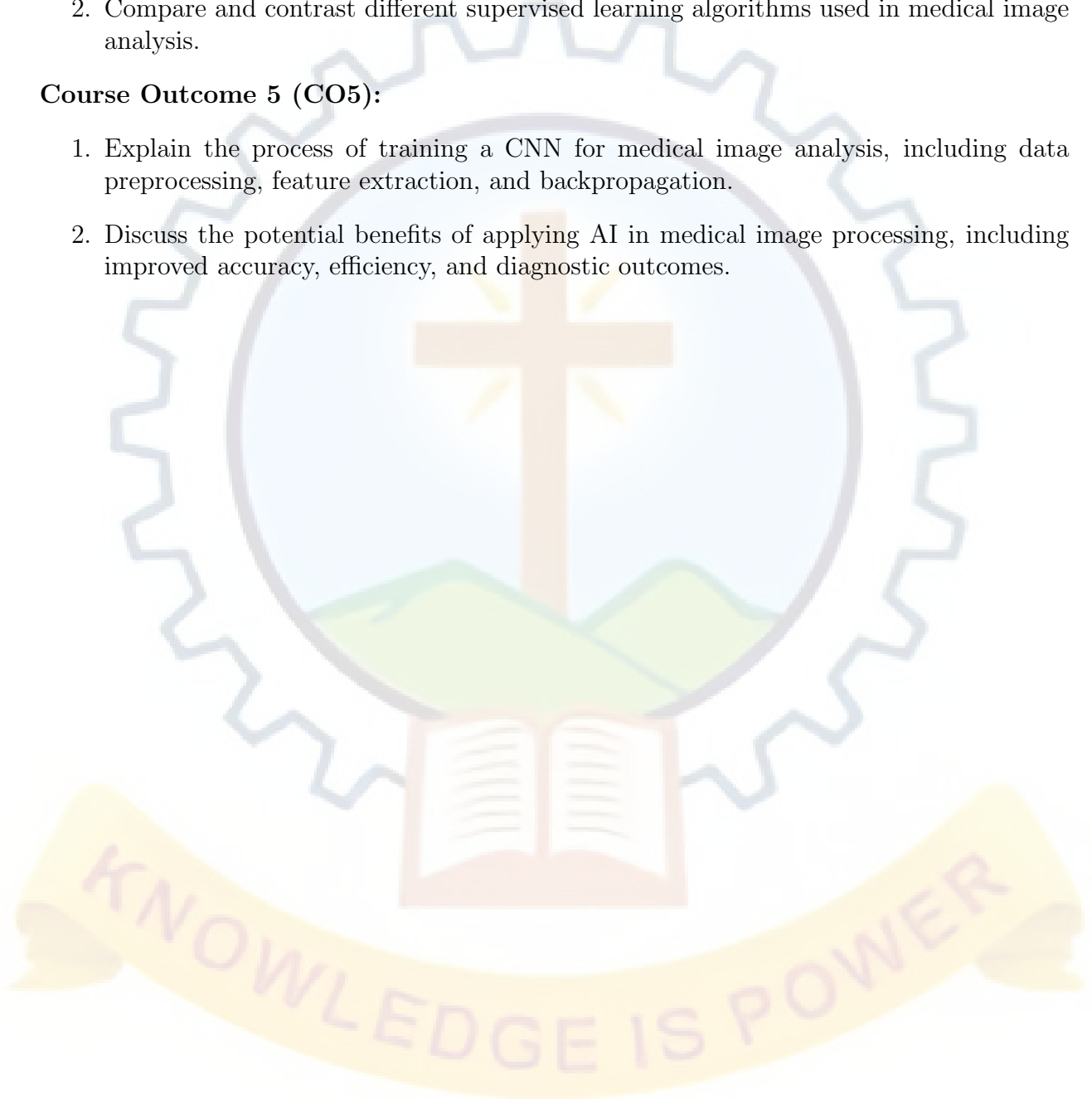
3. Explain the concept of edge detection in medical imaging and its role in image sharpening and feature extraction.

Course Outcome 4 (CO4):

1. Explain the concept of image segmentation and its significance in medical image analysis and diagnosis.
2. Compare and contrast different supervised learning algorithms used in medical image analysis.

Course Outcome 5 (CO5):

1. Explain the process of training a CNN for medical image analysis, including data preprocessing, feature extraction, and backpropagation.
2. Discuss the potential benefits of applying AI in medical image processing, including improved accuracy, efficiency, and diagnostic outcomes.



MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CSH62

Course Name: COMPUTATIONAL HEALTH INFORMATICS (Honours)

Max. Marks: 100

Duration: 3 Hours

PART A

Answer all questions. Each question carries 3 marks.

1. Compare and contrast the techniques, SNOMED CT and ICD.
2. List any three tools commonly used for data visualization in healthcare decision support within the field of Health Informatics with their use.
3. Give examples of specific use cases where blockchain can improve healthcare systems.
4. List any three IoT devices for healthcare with application.
5. Explain the basic principles of Magnetic Resonance Imaging.
6. Specify the different categories of image enhancement techniques used in health informatics.
7. Give examples of different types of medical image segmentation techniques and applications.
8. List any three commonly used supervised and unsupervised learning algorithms for medical image analysis.
9. Draw the architecture of a typical CNN.
10. Give the concept of functional MRI and its applications.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the purpose and role of HL7 standards in healthcare data interoperability. Provide examples of HL7 standards commonly used in clinical settings. (7)
- (b) Discuss the privacy and security concerns related to data capture, storage, and retrieval in health informatics. What are some strategies and best practices to mitigate these concerns and protect patient information? (7)

OR

12. (a) Explain the importance of standardizing health data using controlled vocabularies and classifications like SNOMED CT and ICD. What are the benefits of using standardized codes? (7)
- (b) Discuss the importance of data capture in health informatics. Explain the different methods of data capture used in healthcare settings. (7)
13. (a) Discuss the impact of emerging technologies on health informatics, highlighting their potential benefits and challenges in the healthcare industry. (7)
- (b) Explain the concept of precision medicine and its relationship with translational bioinformatics. How can bioinformatics tools and techniques contribute to the development of personalized treatment approaches? (7)

OR

14. (a) Describe the potential uses of IoT devices in healthcare and discuss their impact on patient care and health monitoring. (7)
- (b) Discuss the types of machine learning algorithms commonly used in healthcare. Provide examples of supervised, unsupervised, and reinforcement learning algorithms and describe their specific applications in healthcare settings. (7)
15. (a) Describe the characteristics of ultrasound waves used in imaging. How does ultrasound utilize sound waves to create images of internal body structures? (7)
- (b) Explain the concept of contrast enhancement in medical image processing. Why is contrast enhancement important in improving the visual quality and diagnostic utility of medical images? (7)

OR

16. (a) Discuss the challenges in medical image analysis posed by the complexity and variability of anatomical structures and diseases. How can these challenges be addressed to improve the accuracy and reliability of image analysis? (7)
- (b) Describe the different types of noise commonly encountered in medical images. Why is it necessary to remove or reduce noise to improve medical images' visual quality and interpretability? (7)

17. (a) Describe the basic principles of supervised learning for classification in medical image analysis. Discuss the steps involved, including data preparation, feature extraction, model training, and model evaluation. (7)
- (b) Discuss the concept of training, validation, and testing datasets in machine learning. Discuss the purpose of each dataset and their roles in evaluating model performance and generalization. (7)

OR

18. (a) How do regression techniques contribute to tasks such as disease prognosis, treatment response prediction, and quantitative analysis in healthcare? (7)
- (b) Discuss the application of edge detection and edge-based features in medical image analysis. List any two edge detection algorithms which can be used to extract edge-based features with their pros and cons. (7)
19. (a) Evaluate the future prospects and advancements in volumetric image analysis and 3D reconstruction in health. Discuss emerging technologies and trends in healthcare. (7)
- (b) Discuss the challenges and considerations in object detection and segmentation using deep learning. (7)

OR

20. (a) Explain the concept of diffusion tensor imaging and its significance in medical imaging. Discuss how diffusion tensor imaging captures and measures the diffusion of water molecules in biological tissues. (7)
- (b) Explain the challenges associated with variability in medical images. Also, explain the challenges of model interpretability and explainability in AI-based medical image processing. (7)

B24CSH63	THEORY OF COMPUTABILITY AND COMPLEXITY (Honours)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		

Preamble

This course provides a deep exploration of the fundamental limits of computation and the classification of problems by their inherent difficulty. This course covers computability theory, NP-completeness, space complexity, and advanced computational models like circuits and randomized computation. Studying this course equips students with rigorous analytical skills and a critical understanding of the theoretical boundaries that underpin all of computer science.

Prerequisites

Automata Theory and Formal Languages, Algorithms and Data Structures, Basic exposure to NP-completeness

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain core computability concepts using Turing Machines and distinguish decidable and undecidable problems.(Cognitive Knowledge Level: Understand)
CO 2	Employ polynomial-time reductions to categorize problems within P, NP, and NP-complete classes.(Cognitive Knowledge Level: Apply)
CO 3	Describe diagonalization techniques and use hierarchy results to compare computational problems.(Cognitive Knowledge Level: Apply)
CO 4	Apply space complexity measures and fundamental results to determine the resource requirements and closure properties of different computational classes. (Cognitive Knowledge Level: Apply)
CO 5	Use advanced computational complexity models and associated analytical techniques/tools to evaluate efficiency and responsible use in computing systems.(Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	2	2		2					2
CO 2	2	2	2	2	1	1					2
CO 3	2	1	2	2		1					2
CO 4	2	1	2	2		1					2
CO 5	2	2	2	2	1	2	1				2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Micro project/Video Assignment/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (9 hours)

Introduction to Computability and Complexity:

Automata, Computability and Complexity- Complexity theory, Computability theory, Automata theory. Review of Turing Machines (TM) – standard model, multi-tape, nondeterministic and enumerator models. Encoding of Turing Machines. Decidability and Recognizability – decidable and semi-decidable languages, recursive and recursively enumerable sets.

MODULE 2 (9 hours)

Time Complexity and NP Completeness:

Time-bounded computation and formal definition of time complexity. Deterministic vs nondeterministic computation. Polynomial-time reducibility and the concept of NP-completeness. Cook–Levin Theorem, examples of NP-complete problems and the P vs NP problem. Additional NP complete problems.

MODULE 3 (8 hours)

Diagonalization and Time Hierarchies:

Diagonalization in computability and complexity – basic idea and examples. Time Hierarchy Theorem – proof intuition and implications. Ladner's Theorem – existence of problems between P and NP-complete. Limits of diagonalization – relativization and natural proof barriers (overview).

MODULE 4 (9 hours)

Space Complexity:

Space-bounded computation and space complexity measures. Classes L, NL, PSPACE and their relationships. NL-completeness, PSPACE-completeness, and Savitch's Theorem. Space Hierarchy Theorem, Immerman–Szelepcsényi Theorem, closure of NL under complement.

MODULE 5 (10 hours)

Advanced Complexity Topics:

Approximation Algorithms, Probabilistic algorithms, Alternation-Alternating time and space, Polynomial time hierarchy. Interactive Proofs - Definition and examples of interactive proofs, $IP = PSPACE$ theorem. Parallel computation-uniform boolean circuits, the class NC, P-completeness.

Text Books

1. Michael Sipser, Introduction to the Theory of Computation, Cengage Learning, 3/e 2014.
2. Christos H. Papadimitriou, Computational Complexity, Addison-Wesley.

Reference Books

1. Sanjeev Arora and Boaz Barak, Computational Complexity: A Modern Approach, Cambridge University Press.
2. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, Pearson.
3. A Oded Goldreich, Computational Complexity: A Conceptual Perspective, Cambridge University Press.
4. Dexter C. Kozen, Automata and Computability, Springer.
5. Thomas H. Cormen, Charles, E. Leiserson, Ronald L. Rivest and Clifford Stein, Introduction to Algorithms, The MIT Press Cambridge 4/e 2023.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Computability and Uncomputability		9 hours
1.1	Automata, Computability and Complexity : Complexity theory	1
1.2	Computability theory	1
1.3	Automata theory	1
1.4	Review of Turing Machines (TM) – standard model, multi-tape	2
1.5	Encoding of Turing Machines	1
1.6	Decidability and Recognizability – decidable and semi-decidable languages	2

1.7	Recursive and recursively enumerable sets.	1
Module 2: Time Complexity and NP-Completeness		9 hours
2.1	Time-bounded computation	1
2.2	Formal definition of time complexity.	1
2.3	Complexity classes P and NP	1
2.4	Deterministic vs nondeterministic computation	1
2.5	Polynomial-time reducibility	1
2.6	The concept of NP-completeness	1
2.7	Cook–Levin Theorem	1
2.8	Examples of NP-complete problems and The P vs NP problem	1
2.9	Additional NP complete problems	1
Module 3: Diagonalization and Time Hierarchies:		8 hours
3.1	Diagonalization in computability – basic idea and examples	1
3.2	Time Hierarchy Theorem – proof intuition and implications.	2
3.3	Ladner’s Theorem – existence of problems between P and NP-complete.	2
3.4	Limits of diagonalization	1
3.5	Relativization barrier (overview)	1
3.6	Natural Proofs barrier (overview)	1
Module 4: Space Complexity		9 hours
4.1	Space-bounded computation	1
4.2	Space complexity measures	1
4.3	Classes L, NL, PSPACE and their relationships.	1
4.4	L-completeness	1
4.5	PSPACE-completeness	1
4.6	Savitch’s Theorem	1
4.7	Space Hierarchy Theorem	1
4.8	Immerman–Szelepcsényi Theorem	1
4.9	Closure of NL under complement.	1
Module 5: Advanced Complexity Topics		10 hours

5.1	Approximation Algorithms	1
5.2	Probabilistic algorithms	2
5.3	Alternation-Alternating time and space	1
5.4	Polynomial time hierarchy	1
5.5	Interactive Proofs - Definition and examples of interactive proofs	1
5.6	IP = PSPACE theorem	1
5.7	Parallel computation-uniform boolean circuits	1
5.8	RP, coRP, ZPP	1
5.9	The class NC	1
5.10	P-completeness.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. “There exists some un-computable problems/function in the Universe”. Is it True or False? Justify with the help of counting and diagonalization method.
2. Consider a nondeterministic Turing Machine N that halts on every input within $O(n^2)$ steps. Construct an equivalent deterministic single-tape Turing Machine D and prove that for every input string w , $N(w) = D(w)$ (i.e., both machines accept or reject the same set of strings).

Course Outcome 2 (CO 2):

1. Let P, Q be two problem then write about any two reduction methods that can reduce problem P to problem Q .
2. Outline the major steps involved in reducing an arbitrary NP problem to SAT.
3. Draw the diagram showing the relation between the complexity classes L , NL , P , NP , $PSPACE$, $CO-NP$ and $CO-NL$.
4. Explain Totally Quantified Boolean Formula (TBQF). Write an Example.

Course Outcome 3 (CO 3):

1. Explain the diagonalization method with an example showing how it is used to separate computational classes.
2. Describe the significance of the Time Hierarchy Theorem. Using its statement, show why more time allows solving strictly more problems.

Course Outcome 4 (CO 4):

1. Explain why $NL \subseteq P$. Illustrate using the space-to-time simulation used in space-bounded computations.
2. Using Savitch's Theorem, derive the deterministic space complexity required to solve ST-CONNECTIVITY. Explain the recursion tree briefly.

Course Outcome 5 (CO 5):

1. A randomized algorithm A solves a decision problem as follows:
With probability $1/2$, it outputs the correct answer "YES" or "NO" in polynomial time.
With probability $1/2$, it outputs "I DON'T KNOW".
If the output is "I DON'T KNOW", the algorithm restarts and runs again using fresh randomness.
 - (a) Show that the expected running time of algorithm A is polynomial.
 - (b) Using the properties of ZPP, determine whether the decision problem solved by A belongs to the class ZPP. Justify your answer.
2. Explain the class BPP and describe how probabilistic algorithms achieve bounded error. Using a suitable example (other than primality), show how a decision problem can be placed in BPP.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2027

Course Code: B24CSH63

*Course Name: THEORY OF COMPUTABILITY AND COMPLEXITY
(Honours)*

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Explain the difference between decidable and semi-decidable languages with one example for each.
2. Describe the concept of a Universal Turing Machine and explain why it is important in computability theory.
3. Define the classes L and NL.
4. Give the definition of PSPACE Completeness
5. Define a problem that is guaranteed to exist between the complexity classes P and NP-complete, as proved by Ladner's Theorem.
6. How is diagonalization technique used to separate different computational models or complexity classes?
7. Explain the concept of Alternating Turing Machines with an example.
8. Distinguish between uniform and non-uniform circuit families.
9. Define the complexity classes RP and ZPP.
10. Explain the concept of bounded error in probabilistic algorithms.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Given a non-deterministic Turing Machine M with $O(n)$ computing steps. If there exist an equivalent deterministic single tape Turing Machine D , then prove that $M(x)=D(x)$ for all input x . (7)
- (b) Write short note on configuration graph of a Turing Machine. (7)

OR

12. (a) Prove that for every multi tape Turing Machine, there exists an equivalent single tape Turing Machine. (7)
- (b) Does there exist a Turing Machine M that can decide the property "Accept a binary string with exactly five bits" on another Turing Machine N ? Justify your answer. (7)
13. (a) Define the class PSPACE. Give one example of a PSPACE-complete problem and briefly explain why it is PSPACE-complete. (7)
- (b) Explain the Polynomial Hierarchy (PH). Describe the classes Σ_k^P and Π_k^P and state the relationship between PH and NP. (7)

OR

14. (a) Let $G=(V,E)$ be a directed graph and . Define the ST-CONNECTIVITY problem formally and explain why it belongs to the class NL. Further, describe the nondeterministic space-bounded algorithm for deciding reachability (7)
- (b) Using Savitch's Theorem, construct a deterministic algorithm to solve ST-CONNECTIVITY and derive the deterministic space bound. Clearly justify why the deterministic version runs in $O((\log n)^2)space$ (7)
15. (a) State and explain Ladner's Theorem. What crucial question regarding the structure of NP does this theorem answer and what is the specific nature of the problems whose existence it proves? (7)
- (b) Illustrate the Natural Proofs barrier using an example and explain why it prevents certain circuit lower bound techniques from proving $P \neq NP$. (7)

OR

16. (a) How do the concepts of relativization barrier and the natural proof barrier represent the limitations of using the diagonalization technique to resolve major open problems in complexity theory, specifically $P = NP$? (7)
- (b) Explain with examples why diagonalization alone is insufficient to resolve the P vs NP problem. (7)
17. (a) Design a randomized polynomial time algorithm for primality testing of an integer number. (7)

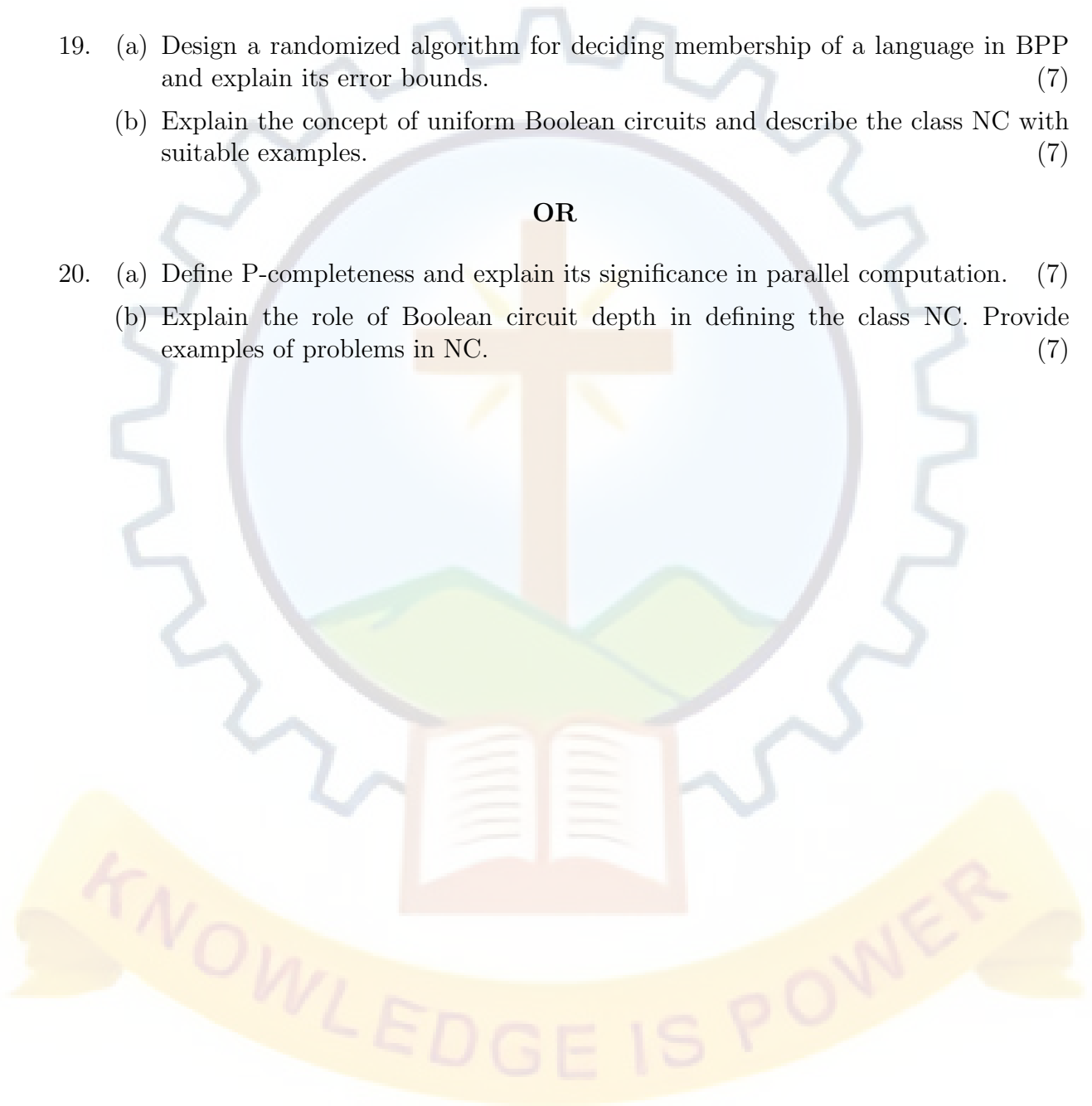
- (b) Write an Interactive proof for graph non-isomorphism (7)

OR

18. (a) Compare the complexity classes RP, coRP, and ZPP with suitable examples. (7)
(b) Demonstrate how error reduction is achieved in randomized algorithms using repetition. (7)
19. (a) Design a randomized algorithm for deciding membership of a language in BPP and explain its error bounds. (7)
(b) Explain the concept of uniform Boolean circuits and describe the class NC with suitable examples. (7)

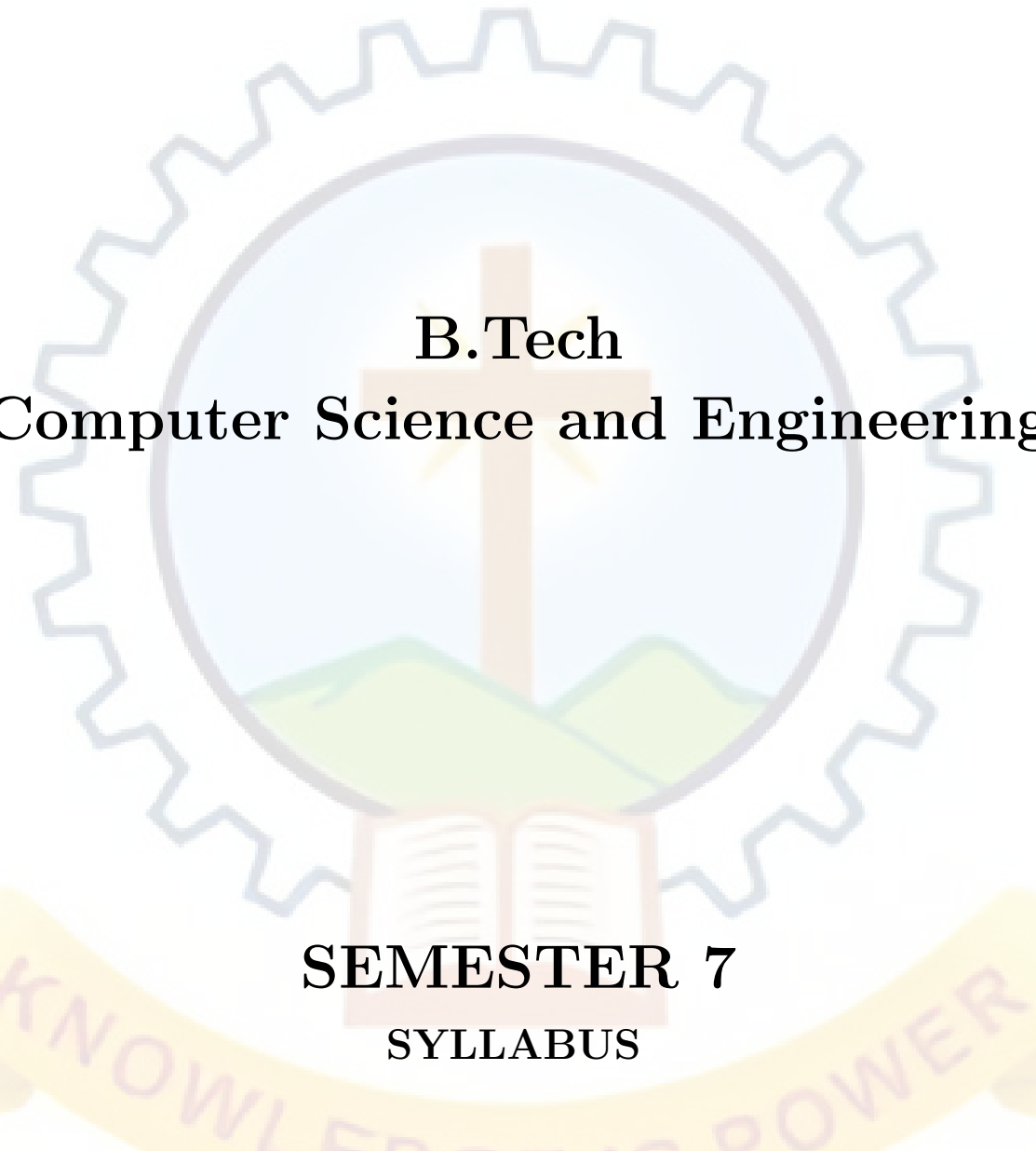
OR

20. (a) Define P-completeness and explain its significance in parallel computation. (7)
(b) Explain the role of Boolean circuit depth in defining the class NC. Provide examples of problems in NC. (7)



MAR ATHANASIOUS COLLEGE OF ENGINEERING

Government Aided, Autonomous Institution
Kothamangalam, Kerala, India

The logo of Mar Athanasius College of Engineering is a circular emblem. It features a large gear-like outer border. Inside the circle, there is a light blue background with a central orange cross. Below the cross, there are green hills. At the bottom of the circle, there is an open book. A yellow banner with the text 'KNOWLEDGE IS POWER' in purple capital letters curves around the bottom of the emblem.

B.Tech Computer Science and Engineering

SEMESTER 7 SYLLABUS

SEMESTER 7

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A	B24CS4T01	BASICS OF MACHINE LEARNING	2-1-0-4	3	3
B	B24CS4P3x	PROGRAMME ELECTIVE III	2-1-0-4	3	3
C	B24CS4P4x	PROGRAMME ELECTIVE IV	2-1-0-4	3	3
D	B24CS4G2x	OPEN ELECTIVE II	2-1-0-4	3	3
E	B24HU4T04	DISASTER MANAGEMENT AND INDUSTRIAL SAFETY	2-1-0-4	3	3
G	B24CS4L11	COMPUTER GRAPHICS LAB	0-0-3-2	3	2
H	B24CS4L12	PROJECT PHASE I	0-0-6-2	6	3
J	B24CS4L13	SEMINAR	0-0-4-1	4	2
K	B24CS4T02	VIVA VOCE	0-0-0-0	-	1
M	B24CSM7x	MINOR	3-1-0-5	4	4
N	B24CSH7x	HONOURS	3-1-0-5	4	4
TOTAL				36	23

PROGRAMME ELECTIVE III

COURSE NO.	COURSES
B24CS4P31	SOFTWARE PERFORMANCE AND SCALABILITY (Bucket -I)
B24CS4P32	INFORMATION AND DATA SECURITY (Bucket -II)
B24CS4P33	HIGH PERFORMANCE COMPUTING (General -I)
B24CS4P34	MODERN DATABASES (General -II)

PROGRAMME ELECTIVE IV

COURSE NO.	COURSES
B24CS4P41	SOFTWARE TESTING ARCHITECTURE (Bucket -I)
B24CS4P42	CYBER FORENSICS (Bucket -II)
B24CS4P43	EMBEDDED SYSTEMS (General -I)
B24CS4P44	CLIENT SERVER COMPUTING (General -II)

OPEN ELECTIVE II

COURSE NO.	COURSES
B24CS4G21	INTRODUCTION TO DATABASES
B24CS4G22	BASICS OF LAN TECHNOLOGIES
B24CS4G23	INTRODUCTION TO MOBILE COMPUTING

HONOURS

COURSE NO.	COURSES
B24CSH71	SURVEILLANCE VIDEO ANALYTICS
B24CSH72	COMPUTER AIDED DRUG DESIGN
B24CSH73	LOGIC FOR COMPUTER SCIENCE

B24CS4T01	BASICS OF MACHINE LEARNING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course enables learners to understand the fundamental and advanced concepts of machine learning and its core algorithms. It covers widely used supervised learning methods such as linear regression, logistic regression, decision trees, Bayesian learning, and the Naive Bayes algorithm, along with essential clustering techniques and model performance evaluation measures. Through this course, students will gain the knowledge and skills required to design and develop machine learning-based solutions for real-world problems.

Prerequisites

Basic knowledge of mathematics (including linear algebra, probability, and statistics), familiarity with programming concepts (preferably Python), and fundamental understanding of data structures and algorithms.

Course Outcomes

After the completion of the course the student will be able to:

CO1	Illustrate the fundamental concepts of Machine Learning, including types of learning, vector space representations, and learning theories such as VC Dimension and PAC Learning. (Cognitive Knowledge Level: Understand)
CO2	Apply data processing techniques using NumPy, Pandas, and Matplotlib for data cleaning, manipulation, and visualization. (Cognitive Knowledge Level: Apply)
CO3	Demonstrate supervised learning techniques including regression and classification, and implement decision tree algorithms for predictive modeling. (Cognitive Knowledge Level: Apply)
CO4	Analyze and apply unsupervised learning techniques such as clustering methods (K-medoids, density-based, and model-based clustering using the EM algorithm). (Cognitive Knowledge Level: Apply)
CO5	Evaluate machine learning models using appropriate experimental design techniques, cross-validation methods, and performance metrics for classification, regression, and clustering. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	1		1	1			2	2		2
CO2	3	2	1	2	2	1		2	2	1	1
CO3	3	2	2	2	2			2	2	2	
CO4	2	2	2	3	3	1		3	2	2	1
CO5	2	2	2	3	2			3		3	2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/ Seminar/ Microproject/ Video Assignments/ Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contains 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 is to be answered. Each question carries 14 marks and can have a maximum of 2 sub-divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction to Machine Learning:

What is Machine Learning? Types of Machine Learning – Supervised Learning, Unsupervised Learning, Reinforcement Learning (overview). Real-world applications of ML. Vector Spaces, inverse and transpose, vector subspace, Linear Independence. Vapnik-Chervonenkis (VC) Dimension, Probably Approximately Correct (PAC) Learning.

MODULE 2 (6 hours)

Statistical Estimation and Dimensionality Reduction:

Maximum Likelihood Estimation- Bernoulli Density - Multinomial Density - Gaussian (Normal) Density, Evaluating an Estimator: Bias and Variance. Dimensionality Reduction- Subset Selection, Principal Components Analysis

MODULE 3 (9 hours)

Supervised Learning & Unsupervised Learning:

Regression- Simple Linear Regression, Multiple Linear Regression, Non Linear Regression. Classification- Logistic Regression, Decision Trees-Classification Trees - ID3, C4.5, CART, Regression Trees. Clustering- Similarity measures. K-medoids clustering, Density based clustering. Model based clustering - Expectation-Maximization Algorithm

MODULE 4 (8 hours)

Design and Analysis of Machine Learning Experiments:

Design and Analysis of Machine Learning Experiments- Guidelines for Machine Learning Experiments, Cross-Validation and Resampling Methods, K-Fold Cross-Validation, 5×2 Cross-Validation, Bootstrapping, Measuring Classifier Performance

MODULE 5 (7 hours)

Performance Measures:

Classification Performance measures - Accuracy, Precision, Recall, Specificity, False Positive Rate (FPR), F1 Score, Receiver Operator Characteristic (ROC) Curve, AUC. Regression Performance Evaluation Metrics: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R Squared/Coefficient of Determination. Clustering Performance Evaluation Metrics: Purity, Jaccard index, Normalized Mutual Information, Clustering Accuracy, Silhouette Coefficient, Dunn's Index.

Text Books

1. Ethem Alpaydin, *Introduction to Machine Learning*, 2nd Edition, MIT Press, 2010.
2. Tom M. Mitchell, *Machine Learning*, McGraw-Hill, 1997.

Reference Books

1. Christopher Bishop, *Neural Networks for Pattern Recognition*, OUP, 1995.
2. Kevin P. Murphy, *Machine Learning: A Probabilistic Perspective*, MIT Press, 2012.
3. Trevor Hastie et al., *The Elements of Statistical Learning*, Springer, 2007.
4. Richert and Coelho, *Building Machine Learning Systems with Python*.
5. Mohammed J. Zaki & Wagner Meira, *Data Mining and Analysis: Fundamental Concepts and Algorithms*, CUP, 2016.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to Machine Learning		7 hours
1.1	What is Machine Learning? Types of ML: Supervised, Unsupervised, Reinforcement Learning	1
1.2	Real-world applications of ML	1
1.3	Vector spaces and basic operations	1

1.4	Inverse, transpose, vector subspace	1
1.5	Linear Independence	1
1.6	VC Dimension	1
1.7	PAC Learning	1
Module 2: Statistical Estimation and Dimensionality Reduction		6 hours
2.1	Maximum Likelihood Estimation (MLE)	1
2.2	Bernoulli Density	1
2.3	Multinomial Density	1
2.4	Gaussian (Normal) Density	1
2.5	Bias and Variance, Subset Selection	1
2.6	Principal Component Analysis (PCA)	1
Module 3: Supervised & Unsupervised Learning		9 hours
3.1	Simple Linear Regression	1
3.2	Multiple Linear Regression	1
3.3	Non-linear Regression	1
3.4	Logistic Regression	1
3.5	Decision Trees: ID3, C4.5	1
3.6	CART and Regression Trees	1
3.7	Similarity Measures in Clustering, K-medoids Clustering	1
3.8	Density-based Clustering (DBSCAN)	1
3.9	EM Algorithm (Model-based Clustering)	1
Module 4: Design and Analysis of ML Experiments		6 hours
4.1	Guidelines for ML Experiments	1
4.2	Cross-validation concepts	1
4.3	K-Fold Cross-validation	1
4.4	5×2 Cross-validation	1
4.5	Bootstrapping	1
4.6	Measuring Classifier Performance	1
Module 5: Performance Measures		8 hours
5.1	Accuracy, Precision, Recall	1
5.2	Specificity and False Positive Rate (FPR)	1

5.3	F1 Score	1
5.4	ROC Curve	1
5.5	AUC, Regression Metrics: MAE, RMSE, R^2	2
5.6	Clustering Metrics: Purity, Jaccard Index	1
5.7	NMI, Silhouette Coefficient, Dunn's Index	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain different types of Machine Learning (supervised, unsupervised, and reinforcement learning) with suitable examples.
2. Discuss vector space concepts including linear independence and subspaces. Explain their significance in Machine Learning.
3. Explain the concepts of VC Dimension and PAC Learning and their role in learning theory.

Course Outcome 2 (CO2):

1. A coin is tossed 100 times and lands heads 62 times. Estimate the probability of heads using Maximum Likelihood Estimation.
2. Derive the Maximum Likelihood Estimator (MLE) for the mean of a Gaussian distribution with known variance.
3. Explain bias and variance. Discuss their impact on model performance.
4. Apply Principal Component Analysis (PCA) to reduce dimensionality and explain how variance is preserved.

Course Outcome 3 (CO3):

1. Derive the equation of simple linear regression and explain its assumptions.
2. Explain logistic regression for binary classification and its decision boundary.
3. Construct a decision tree using ID3 algorithm for a given dataset.

4. Compare ID3, C4.5, and CART algorithms.

Course Outcome 4 (CO4):

1. Explain similarity measures used in clustering with examples.
2. Describe the K-medoids clustering algorithm and compare it with K-means.
3. Explain density-based clustering (DBSCAN) and its advantages.
4. Describe the Expectation-Maximization (EM) algorithm for model-based clustering.

Course Outcome 5 (CO5):

1. Explain K-fold cross-validation, 5×2 cross-validation, and bootstrapping techniques.
2. Evaluate a classification model using metrics such as accuracy, precision, recall, F1-score, and ROC curve.
3. Explain regression evaluation metrics: MAE, RMSE, and R^2 .
4. Discuss clustering evaluation metrics such as Silhouette coefficient and Dunn's Index.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CS4T01

Course Name: BASICS OF MACHINE LEARNING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define supervised, unsupervised and reinforcement learning with one example each.
2. What is linear independence? Give an example.
3. Define VC Dimension and state its significance.
4. What is PAC Learning? Explain briefly.
5. A coin is tossed 100 times and lands heads 62 times. Estimate the probability using MLE.
6. Define bias and variance.
7. Write the formula for Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE).
8. What is K-medoids clustering?
9. Define K-fold cross-validation.
10. A classifier has $TP=40$, $FP=10$, $FN=5$. Compute Precision and Recall.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain types of Machine Learning with suitable examples. (8)
(b) Discuss vector spaces, subspaces, and linear independence with examples. (6)

OR

12. (a) Explain VC Dimension and its importance in learning theory. (8)
(b) Describe PAC Learning framework with suitable explanation. (6)
13. (a) Derive the Maximum Likelihood Estimator for Bernoulli distribution. (8)
(b) Explain Gaussian distribution and its parameters. (6)

OR

14. (a) Explain bias-variance trade-off with suitable examples. (8)
(b) Describe Principal Component Analysis (PCA) and its role in dimensionality reduction. (6)
15. (a) Derive simple linear regression model and explain assumptions. (8)
(b) Explain multiple linear regression and its challenges. (6)

OR

16. (a) Explain logistic regression and decision boundary. (8)
(b) Compare ID3, C4.5, and CART algorithms. (6)
17. (a) Explain K-fold cross-validation and 5×2 cross-validation. (8)
(b) Describe bootstrapping technique. (6)

OR

18. (a) Explain guidelines for designing Machine Learning experiments. (8)
(b) Discuss methods for measuring classifier performance. (6)
19. (a) Explain Accuracy, Precision, Recall, and F1-score. (8)
(b) Describe ROC curve and AUC. (6)

OR

20. (a) Explain regression evaluation metrics: MAE, RMSE, and R^2 . (8)
(b) Discuss clustering evaluation metrics: Silhouette coefficient and Dunn's Index. (6)

B24CS4P31	SOFTWARE PERFORMANCE AND SCALABILITY	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course provides a comprehensive foundation in software performance and scalability. It examines hardware platforms, including computational models proposed by Alan Turing and John von Neumann, alongside modern processor architectures. The syllabus addresses software architectures, APIs, enterprise systems, and multithreading concepts. The course prepares students to design software that is fast, efficient, reliable, and scalable in real-world environments.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	To explain and analyze fundamental hardware platforms including Turing, von Neumann, Zuse, Intel, and Sun machines, and evaluate modern processor architectures, and hardware sizing considerations. (Cognitive Knowledge Level: Understand)
CO 2	To describe and apply software platform concepts such as software stacks, APIs (Windows, Java, Google), multithreading, and classify system, application, middleware, and enterprise software. (Cognitive Knowledge Level: Apply)
CO 3	To apply and analyze various software performance testing techniques and evaluate the role of performance testing in software development processes including Agile and Extreme Programming (Cognitive Knowledge Level: Apply)
CO 4	To interpret and apply queuing theory concepts, metrics, probability theory, random variables, and discrete and continuous distributions to model, analyze, and predict system performance and scalability behavior using mathematical representations. (Cognitive Knowledge Level: Apply)
CO 5	To implement an API profiling framework, analyze execution stacks and performance logs, and create performance maps and summarization reports to identify and defend against software performance and scalability defects. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1		1	1				1		1
CO 2	2	3	3	3	3	2		2	3	2	3
CO 3	2	3	3	3	3	3	3	3	3	3	3
CO 4	2	3	1	3	2				1		3
CO 5	2	3	3	3	3	3	3	3	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	15	15	15
Understand	55	55	55
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/Seminar/Microproject/Video Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

HARDWARE PLATFORM

THE BASICS , Hardware Platform , Turing Machine, von Neumann Machine, Zuse Machine , Intel Machine , History of Intel's Chips , Hyperthreading , Intel's Multicore Microarchitecture , Challenges for System Monitoring Tools , Sun Machine , System Under Test , Processors , Motherboard , Chipset , Storage , RAID , Networking , Operating System ,Odds Against Turing , Memory Leaks , SLAs, Sizing Hardware .

MODULE 2 (7 hours)

SOFTWARE PLATFORM

Software Stack , APIs , Windows APIs , Java APIs ,Google APIs , Multithreading, Categorizing Software , Systems Software, Application Software ,Middleware Software , Enterprise Computing ,What Is Enterprise Software? Enterprise Software Architecture ,Monolithic Architecture,Client /Server Architecture , Three-Tier Architecture ,N-Tier Architecture , Software Componentry , Service-Oriented Architecture.

MODULE 3 (7 hours)

TESTING SOFTWARE PERFORMANCE AND SCALABILITY

Scope of Software Performance and Scalability Testing , Performance Regression Testing ,Performance Optimization and Tuning Testing , Performance Benchmarking Testing , Scalability Testing , QA Testing Versus Performance Testing , Additional Merits of Performance Testing , Software Development Process , Agile Software Development ,Extreme Programming

MODULE 4 (8 hours)

APPLYING QUEING THEORY

Introduction to Queuing Theory , Queuing Concepts and Metrics , Basic Concepts of Queuing Theory , Queuing Theory: From Textual Description to Mathematical Symbols

, Introduction to Probability Theory , Random Variables and Distribution Functions , Discrete Distribution and Probability, Distribution Series, Continuous Distribution and Distribution Density Function.

MODULE 5 (7 hours)

APPLYING API PROFILING

Defining API Profiling Framework , Defense Lines Against Software Performance and Scalability Defects , Software Program Execution Stack , The PerfBasic API Profiling Framework , API Profile Logging Format, Performance Log Parser , Performance Maps , Performance Summarization File.

Text Books

1. Henry H. Liu "Software Performance aND Scalability-A Quantitative Approach" Wiley-Blackwell (Wiley) 1st Ed (2009)
2. Kai Hwang, Naresh Jotwani,"Advanced Computer Architecture" McGraw-Hill Education 3rd Ed (2015)

Reference Books

1. Andre B Bonde, "Foundations of Software and System Performance Engineering",Pearson Education, 4th Edition, 2011
2. Alon Rotem, "Software Performance Engineering: A comprehensive guide for high-performance development", BPB Publications, 2025

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1:HARDWARE PLATFORM		7 hours
1.1	THE BASICS , Hardware Platform , Turing Machine, von Neumann Machine	1
1.2	Zuse Machine , Intel Machine , History of Intel's Chips , Hyperthreading	1
1.3	Intel's Multicore Microarchitecture , Challenges for System Monitoring Tools	1

1.4	Tun Machine , System Under Test , Processors , Motherboard	1
1.5	Chipset , Storage , RAID	1
1.6	Networking , Operating System	1
1.7	Odds Against Turing , Memory Leaks , SLAs, Sizing Hardware	1
Module 2:SOFTWARE PLATFORM		7 hours
2.1	Software Stack , APIs , Windows APIs , Java APIs ,Google APIs , Multithreading	1
2.2	Categorizing Software , Systems Software, Application Software ,Middleware Software	1
2.3	Enterprise Computing ,What Is Enterprise Software? Enterprise Software Architecture	1
2.4	Monolithic Architecture,,Client /Server Architecture	1
2.5	Three-Tier Architecture ,N-Tier Architecture	1
2.6	Software Componentry	1
2.7	Service-Oriented Architecture	1
Module 3:TESTING SOFTWARE PERFORMANCE AND SCALABILITY		7 hours
3.1	Scope of Software Performance and Scalability Testing	1
3.2	Performance Regression Testing ,Performance Optimization and Tuning Testing	1
3.3	Performance Benchmarking Testing , Scalability Testing	1
3.4	QA Testing Versus Performance Testing , Additional Merits of Performance Testing	1
3.5	Software Development Process	1
3.6	Agile Software Development	1
3.7	Extreme Programming	1
Module 4:APPLYING QUEING THEORY		8 hours
4.1	Introduction to Queuing Theory , Queuing Concepts and Metrics	2
4.2	Basic Concepts of Queuing Theory	1
4.3	Queuing Theory: From Textual Description to Mathematical Symbols	1
4.4	Introduction to Probability Theory	1

4.5	Random Variables and Distribution Functions	1
4.6	Discrete Distribution and Probability	1
4.7	Distribution Series	1
4.8	Continuous Distribution and Distribution Density Function	1
Module 5:APPLYING API PROFILING		7 hours
5.1	Defining API Profiling Framework	1
5.2	Defense Lines Against Software Performance and Scalability Defects	1
5.3	Software Program Execution Stack	1
5.4	The PerfBasic API Profiling Framework	1
5.5	API Profile Logging Format,	1
5.6	Performance Log Parser	1
5.7	Performance Maps, Performance Summarization File	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Analyze how Hyperthreading and multicore microarchitecture improve system performance, and evaluate their impact on system monitoring tools.
2. Compare and assess the architectural differences between the Turing Machine, von Neumann Machine, and modern Intel machines in terms of practical computing limitations.

Course Outcome 2 (CO 2):

1. Analyze the role of APIs (Windows, Java, Google) in building scalable enterprise software systems.
2. Evaluate monolithic, client-server, and N-tier architectures for enterprise computing, justifying which architecture best supports scalability and maintainability

Course Outcome 3 (CO 3):

1. Analyze the importance of performance regression testing in Agile and Extreme Programming environments

2. Evaluate the differences between performance benchmarking and scalability testing with respect to system growth and workload variation.

Course Outcome 4 (CO 4):

1. Analyze how probability distributions and random variables help in predicting system response time and throughput.
2. Evaluate the transformation of a real-world system description into mathematical queuing models and discuss its advantages for performance analysis.

Course Outcome 5 (CO 5):

1. Given a software program execution stack and API profile logging data, analyze the logs using a performance log parser and interpret the resulting performance maps to locate performance bottlenecks.
2. Design and justify a workflow using the PerfBasic API Profiling Framework to generate performance maps and a performance summarization file for large-scale software systems.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CS4P31

Course Name: SOFTWARE PERFORMANCE AND SCALABILITY

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define a hardware platform and list any two key components of it.
2. Differentiate between a Turing Machine and a von Neumann Machine.
3. What is Hyperthreading? State its advantage in Intel processors.
4. Define a software stack with a suitable example.
5. What are APIs? Mention any two examples of commonly used APIs.
6. Differentiate between system software and application software.
7. Define performance regression testing.
8. What is scalability testing? Why is it important for enterprise systems?
9. Define queuing theory and list any two queuing metrics.
10. What is API profiling? State its importance in software performance analysis.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the evolution of computing machines from the Zuse Machine to modern Intel multicore architectures. (8)
- (b) Discuss the challenges faced by system monitoring tools in multicore and multithreaded systems. (6)

OR

12. (a) Explain the architecture and working of a Sun Machine and compare it with an Intel Machine. (8)
 - (b) Discuss the role of SLAs and hardware sizing in system performance evaluation. (6)
13. (a) Explain enterprise computing and describe the characteristics of enterprise software. (6)
 - (b) Compare monolithic architecture and client-server architecture with respect to scalability and maintenance. (8)

OR

14. (a) Explain three-tier and N-tier architectures with suitable diagrams. (8)
 - (b) Discuss the role of middleware software and multithreading in enterprise applications (6)
15. (a) Explain the scope of software performance and scalability testing (6)
 - (b) Compare performance benchmarking testing and performance optimization and tuning testing. (8)

OR

16. (a) Differentiate between QA testing and performance testing. (6)
 - (b) Explain the role of Agile Software Development and Extreme Programming in performance testing. (8)
17. (a) Explain basic queuing concepts and metrics used in system performance analysis. (7)
 - (b) Describe how a real-world system can be converted from a textual description to a mathematical queuing model. (7)

OR

18. (a) Explain random variables and distribution functions with suitable examples. (7)
 - (b) Differentiate between discrete and continuous probability distributions. (7)
19. (a) Explain the components of an API profiling framework and its role in detecting performance defects (6)
 - (b) Describe the PerfBasic API Profiling Framework and explain its working. (8)

OR

20. (a) Explain the software program execution stack and its importance in performance analysis (6)
- (b) Describe the process of generating performance maps and a performance summarization file using API profile logs. (8)

B24CS4P32	INFORMATION AND DATA SECURITY	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course provides a comprehensive foundation in information system security, covering principles, processes, and technologies required to protect modern computing environments. It progresses from core security concepts and access control mechanisms to practical security concerns in operating systems, web applications, e-mail, and DNS infrastructure. The course concludes with covert communication techniques, enabling learners to understand both defensive strategies and emerging challenges in securing information systems across the system development life cycle.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain security principles, system lifecycle processes, and risk management techniques in information systems. (Cognitive Knowledge Level: Understand)
CO 2	Identify different access control models and authentication mechanisms for secure system design. (Cognitive Knowledge Level: Understand)
CO 3	Describe common web and client-side attacks and propose secure coding and mitigation strategies. (Cognitive Knowledge Level: Apply)
CO 4	Assess security issues in e-mail and DNS systems and recommend appropriate protection mechanisms. (Cognitive Knowledge Level: Apply)
CO 5	Explain and evaluate information hiding techniques such as steganography and digital watermarking, including their security implications. (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	3	2	3	3	2	2			2	2
CO 2	3	3	2	3	3	2	2			2	2
CO 3	3	3	2	3	3	2	2			2	2
CO 4	2	3	2	3	3	2	2			2	2
CO 5	3	3	2	3	3	2	2			2	2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	15	15	15
Understand	55	55	55
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/ Seminar/ Microproject/ Video Assignments/ Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Information System Security Principles

Key Principles of Network Security, Formal Processes - The systems engineering process, The Information Assurance Technical Framework, The Information Systems Security Engineering process, The Systems Development Life Cycle, Information systems security and the SDLC, Risk management and the SDLC.

MODULE 2 (7 hours)

Access Control Considerations

Control Models – DAC, MAC – Types of Access Control Implementations, Identification and Authentication. Remote Access – RADIUS, Password Authentication Protocol, Challenge Handshake Authentication Protocol. Unix/Linux Security – Physical security, Controlling the configuration – Kernel configurations, controlling processes, Controlling users, Encryption and certificates.

MODULE 3 (7 hours)

Web Browser and Client Security

Web Browser and Client Risk – Working of a Web Browser – Web Browser Attacks – Hijacking attack, Replay Attack, Browser parasites – Web Browser Configurations. Web Security - HTTP implementation – Server Content – Client Content – State – Tracking state – Cookies, Web bugs URL tracking, Attacking Web Servers – Account harvesting, SQL injection

MODULE 4 (8 hours)

E-MAIL and DNS Security

The E-mail Risk, Data vulnerabilities: e-mail confidentiality, integrity, availability issues, The E-mail Protocols – SMTP, POP, E-mail Authentication. Domain Name System: Security Issues with DNS – DNS Attacks – Designing DNS – Master Slave DNS Detailed DNS Architecture.

MODULE 5 (7 hours)

Covert Communication

Covert Communication – Steganography – Principles of Steganography – Steganography Compared to Cryptography – Types of Steganography. Steganography Versus Digital Watermarking - Properties of digital watermarking - Types of Digital Watermarking

Text Books

1. Eric Cole, Ronald Krutz, James W. Conley, "Network Security Bible", First Edition Wiley India Pvt Ltd, 2010
2. Michael A Whitman, Herbert J.Mattord, "Principles of Information Security", Cengage Learning, Fourth Edition, 2016.

Reference Books

1. William Stallings, "Network Security Essentials", Pearson Education, 4th Edition, 2011
2. Eric Maiwald, "Fundamentals of Network Security", Tata McGraw-Hill, 2011

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Information System Security Principles		7 hours
1.1	Key Principles of Network Security	1
1.2	Formal Processes - The systems engineering process,	1
1.3	The Information Assurance Technical Framework	1
1.4	The Information Systems Security Engineering process	1
1.5	The Systems Development Life Cycle	1
1.6	Information systems security and the SDLC	1
1.7	Risk management and the SDLC.	1
Module 2: Access Control Considerations		7 hours
2.1	Control Models – DAC, MAC – Types of Access Control Implementations	1
2.2	Identification and Authentication	1
2.3	Remote Access – RADIUS, Password Authentication Protocol	1
2.4	Challenge Handshake Authentication Protocol.	1
2.5	Unix/Linux Security – Physical security,	1
2.6	Controlling the configuration – Kernel configurations , controlling processes	1

2.7	Controlling users, Encryption and certificates.	1
Module 3:Web Browser and Client Security		7 hours
3.1	Web Browser and Client Risk – Working of a Web Browser	1
3.2	Web Browser Attacks – Hijacking attack, Replay Attack, Browser parasites	1
3.3	Web Browser Configurations.	1
3.4	Web Security - HTTP implementation – Server Content – Client Content	1
3.5	State – Tracking state – Cookies, Web bugs URL tracking	1
3.6	Attacking Web Servers – Account harvesting,	1
3.7	SQL injection	1
Module 4: E-MAIL and DNS Security		8 hours
4.1	The E-mail Risk, Data vulnerabilities	2
4.2	e-mail confidentiality, integrity, availability issues,	1
4.3	The E-mail Protocols – SMTP, POP, E-mail Authentication	1
4.4	Domain Name System	1
4.5	Security Issues with DNS – DNS Attacks	1
4.6	Designing DNS	1
4.7	Master Slave DNS	1
4.8	Detailed DNS Architecture.	1
Module 5:Covert Communication		7 hours
5.1	Covert Communication – Steganography	1
5.2	Principles of Steganography	1
5.3	Steganography Compared to Cryptography .	1
5.4	Types of Steganography & Lifting	1
5.5	Steganography Versus Digital Watermarking	1
5.6	Properties of digital watermarking	1
5.7	Types of Digital Watermarking	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the key principles of information system and network security. Discuss how formal processes like the Systems Engineering Process and the Information Assurance Technical Framework help in building secure information systems.
2. Describe the relationship between Information Systems Security Engineering (ISSE) and the Systems Development Life Cycle (SDLC). Describe how risk management is integrated into different phases of the SDLC.

Course Outcome 2 (CO 2):

1. Compare Discretionary Access Control (DAC) and Mandatory Access Control (MAC). Explain different types of access control implementation with suitable examples.
2. Explain identification and authentication mechanisms. Describe remote access authentication methods such as RADIUS, PAP, and CHAP, and discuss their role in securing UNIX/Linux systems.

Course Outcome 3 (CO 3):

1. Explain how a web browser works and discuss common web browser risks and attacks such as hijacking attacks, replay attacks, and browser parasites.
2. Describe web security mechanisms with respect to HTTP implementation. Explain cookies, web bugs, and URL tracking, and discuss common web server attacks like SQL injection and account harvesting.

Course Outcome 4 (CO 4):

1. Explain the risks associated with e-mail systems. Discuss vulnerabilities in e-mail data related to confidentiality, integrity and availability and explain the working of the SMTP and POP protocols.
2. Describe the security issues related to the Domain Name System (DNS). Explain common DNS attacks and discuss the design of DNS including master-slave architecture.

Course Outcome 5 (CO 5):

1. Explain the concept of covert communication. Describe the principles and types of steganography and compare steganography with cryptography.
2. Differentiate between steganography and digital watermarking. Explain the properties and types of digital watermarking and discuss their applications in information security.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CS4P32

Course Name: INFORMATION AND DATA SECURITY

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define information system security and list any two key principles of network security.
2. What is the Systems Development Life Cycle (SDLC)? Mention the importance of security in SDLC.
3. Differentiate between Discretionary Access Control (DAC) and Mandatory Access Control (MAC).
4. What is authentication? Name any two remote access authentication protocols.
5. What is a web browser? Mention any two web browser risks.
6. Define replay attack and browser parasites.
7. List any three e-mail data vulnerabilities.
8. What is DNS? Mention any two DNS attacks
9. Define covert communication and steganography.
10. What is digital watermarking? Mention any two properties of digital watermarking.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. Explain the key principles of network security. Discuss the Systems Engineering Process and the Information Assurance Technical Framework in detail. (14)

OR

12. Describe the Information Systems Security Engineering (ISSE) process. Explain how risk management is integrated into different phases of the SDLC. (14)
13. Explain access control models DAC and MAC. Describe different types of access control implementations with suitable examples. (14)

OR

14. Explain UNIX/Linux security mechanisms including physical security, kernel configuration, user control, encryption, certificates, and UNIX hardening. (14)
15. Explain the working of a web browser. Discuss web browser attacks such as hijacking attacks, replay attacks, and browser parasites. (14)

OR

16. Explain web security with respect to HTTP implementation, cookies, web bugs, URL tracking, and common web server attacks such as SQL injection and account harvesting. (14)
17. Explain e-mail security issues related to confidentiality, integrity, and availability. Describe SMTP and POP protocols and e-mail authentication methods. (14)

OR

18. Explain DNS security issues in detail. Describe DNS architecture including master-slave DNS and discuss common DNS attacks (14)
19. Explain the principles and types of steganography. Compare steganography with cryptography. (14)

OR

20. Differentiate steganography and digital watermarking. Explain the types and applications of digital watermarking. (14)

B24CS4P33	HIGH PERFORMANCE COMPUTING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4		2024

Preamble

This course aims to provide students with a strong foundation in modern processor architectures, memory hierarchies, and parallel computing concepts essential for high-performance computing. The course content includes performance evaluation, serial code optimization techniques, shared-memory programming using OpenMP, and distributed-memory programming using MPI. By studying this course, students will gain the ability to analyze system performance and develop efficient, scalable parallel programs for real-world computational and engineering applications.

Prerequisite

Computer Organization and Architecture.

Course Outcomes

After the completion of the course the student will be able to:

CO1	Analyze modern processor architectures including multicore, multithreaded, and vector processors to evaluate performance characteristics and identify computational bottlenecks. (Cognitive Knowledge Level: Apply)
CO2	Apply serial code optimization techniques such as function inlining, loop transformations, and common subexpression elimination to improve program efficiency. (Cognitive Knowledge Level: Apply)
CO3	Develop and evaluate shared-memory parallel applications using OpenMP while analyzing synchronization issues and performance challenges such as false sharing. (Cognitive Knowledge Level: Apply)
CO4	Design and implement distributed-memory parallel solutions using MPI by applying efficient communication strategies, performance optimization techniques, and domain decomposition methods, including collective operations and nonblocking communication. (Cognitive Knowledge Level: Apply)
CO5	Analyze, design, and implement efficient parallel computing solutions by integrating appropriate architectures, programming models, and performance evaluation techniques to solve real-world computational problems. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	1		1	1	1				1	
CO 2	2	1		1	1						
CO 3	2	1	1	1	1	1				1	
CO 4	2	1	1	1	1	1		1	1	1	
CO 5	2	1	1	1	1	3	2	2	1	1	1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study (Seminar/Quiz/ micro Project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE-1 (7 hours)

Modern Processors and Memory Hierarchy: Stored-program computer architecture, General-purpose cache-based microprocessor architecture. Performance metrics and benchmarks (STREAM failure reasons). Moore's Law, Pipelining, Superscalarity, and SIMD. Multicore and Multithreaded processors. Vector processors: Design principles

MODULE-2 (7 hours)

Serial Code Optimization and Parallel Paradigms: Basic optimization techniques for serial code: Scalar profiling (function-level and line-level), elimination of common subexpressions, avoiding branches, compiler logs. Taxonomy of parallel computing paradigms: Shared-memory computers (UMA, ccNUMA), Distributed-memory computers, and Hierarchical systems. Interconnection Networks: Buses, Switched networks, Fat-tree, and Mesh networks.

MODULE-3 (8 hours)

Shared Memory Programming with OpenMP: Introduction to OpenMP: Parallel execution model, compiler directives, and data scoping (Private, Shared, Firstprivate). Work-sharing constructs: For-loops, sections, and tasking. Synchronization: Critical sections, atomics, and reductions. Case Study: Implementation and Parallelization of Jacobi's Algorithm.

MODULE-4 (8 hours)

Distributed Memory Programming with MPI: Message Passing Interface (MPI) model: Process management and point-to-point communication (Blocking vs. Non-blocking). Collective communication: Broadcast, Scatter, Gather, and Reduction operations. Virtual topologies: Cartesian topologies and `MPI_Cart_create()`. Performance issues: Serialization, contention, and message aggregation. Parallel Scalability: Amdahl's Law and Gustafson's Law.

MODULE-5 (6 hours)

Efficient MPI programming : MPI performance tools- communication parameters- Synchronization, serialization, contention- Reducing communication overhead- optimal domain decomposition. Aggregating messages – Nonblocking Vs Asynchronous communication- Collective communication. Intra-node point-to-point communication.

Text Books

1. Georg Hager, Gerhard Wellein, “Introduction to High Performance Computing for Scientists and Engineers”, Chapman & Hall / CRC Computational Science series, 2011.
2. Charles Severance, Kevin Dowd, “High Performance Computing”, O’Reilly Media, 1998.

Reference Books

1. Kai Hwang, “Advanced Computer Architecture”, McGraw Hill, 2008.
2. Peter Pacheco, “An Introduction to Parallel Programming”, Morgan Kaufmann, 2011.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Modern Processors and Memory Hierarchy		7 hours
1.1	Stored-Program Computer Architecture	1
1.2	Cache-Based Microprocessor Architecture	1
1.3	Performance Metrics and Benchmarks (STREAM)	1
1.4	Moore’s Law and Pipelining	1
1.5	Superscalar and SIMD Architectures	1
1.6	Multicore and Multithreaded Processors	1
1.7	Vector Processors – Design Principles	1
Module 2: Serial Code Optimization and Parallel Paradigms		7 hours
2.1	Introduction to Serial Code Optimization	1
2.2	Scalar Profiling – Function-Level and Line-Level	1

2.3	Common Subexpression Elimination and Branch Avoidance	1
2.4	Compiler Logs and Optimization Insights	1
2.5	Parallel Computing Paradigms – UMA and ccNUMA	1
2.6	Distributed and Hierarchical Systems	1
2.7	Interconnection Networks – Bus, Fat-tree, Mesh	1
Module 3: Shared Memory Programming with OpenMP		8 hours
3.1	Introduction to OpenMP and Execution Model	1
3.2	Compiler Directives and Data Scoping	1
3.3	Work-Sharing Constructs – For Loops	1
3.4	Sections and Tasking	1
3.5	Synchronization – Critical Sections	1
3.6	Atomics and Reduction Operations	1
3.7	Performance Considerations in OpenMP	1
3.8	Case Study: Jacobi Algorithm Parallelization	1
Module 4: Distributed Memory Programming with MPI		8 hours
4.1	MPI Model and Process Management	1
4.2	Point-to-Point Communication – Blocking	1
4.3	Non-Blocking Communication	1
4.4	Collective Communication – Broadcast, Scatter	1
4.5	Gather and Reduction Operations	1
4.6	Virtual Topologies – MPI_Cart_create()	1
4.7	Performance Issues – Serialization and Contention	1
4.8	Scalability – Amdahl's Law and Gustafson's Law	1
Module 5: Efficient MPI Programming		6 hours
5.1	MPI Performance Tools and Communication Parameters	1
5.2	Synchronization, Serialization, and Contention	1
5.3	Reducing Communication Overhead	1
5.4	Optimal Domain Decomposition	1
5.5	Message Aggregation and Nonblocking Communication	1
5.6	Intra-node Point-to-Point Communication	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. List the reasons for the failure of a benchmark like STREAM in evaluating cache-based systems.
2. Illustrate the working of a vector processor with a block diagram and a sample program for array addition.
3. Explain the concept of Pipelining and the Stored Program Computer Architecture.

Course Outcome 2 (CO2):

1. Explain function inlining with an example and explain how it improves performance.
2. A given C program contains redundant computations and frequent branching inside loops. Apply suitable serial code optimization techniques such as common subexpression elimination and branch avoidance to improve its performance. Explain the optimized version.
3. Distinguish between UMA and ccNUMA memory architectures.

Course Outcome 3 (CO3):

1. Write an efficient OpenMP program to add the first 100 numbers assuming 100 available cores.
2. Demonstrate "False Sharing" with a code example and explain its impact on cache performance.
3. Explain how synchronization is handled in OpenMP using the `reduction` clause.

Course Outcome 4 (CO4):

1. Write an MPI program to broadcast an integer value from a root process to all other processes.
2. Compare Blocking and Non-blocking point-to-point communication in MPI.
3. State Amdahl's Law. Why can the serial part of a code never be fully eliminated?
4. Explain the significance of virtual topologies and the `MPI_Cart_create()` function.

Course Outcome 5 (CO5):

1. Explain the role of MPI performance analysis tools in identifying communication bottlenecks.
2. Explain message aggregation in MPI.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CS4P33

Course Name: HIGH PERFORMANCE COMPUTING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Explain function inlining with an example.
2. Why does the STREAM benchmark often fail on modern cache-based processors?
3. Define Superscalarity in modern processors.
4. Differentiate between UMA and NUMA architectures.
5. What is the purpose of the `firstprivate` clause in OpenMP?
6. Define False Sharing and its impact on performance.
7. Explain the `MPI_Gather` operation.
8. State Amdahl's Law.
9. What is domain decomposition in parallel computing?
10. Mention two advantages of non-blocking communication in MPI.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Illustrate the working of a vector processor with a block diagram. (7)
(b) Explain different cache mapping methods and their impact on performance. (7)

OR

12. (a) Discuss the architecture of Multithreaded processors. (7)
(b) Explain Moore's Law and its implications for HPC benchmarks. (7)
13. A given C program contains redundant computations and frequent branching inside loops. Apply suitable serial code optimization techniques such as common subexpression elimination and branch avoidance to improve its performance. Explain the optimized version. (7)
(b) Compare Switched networks and Fat-tree networks in HPC clusters. (7)

OR

14. (a) Explain scalar profiling techniques at the function and line levels. (7)
(b) Describe the UMA and ccNUMA memory paradigms. (7)
15. (a) Write an OpenMP program to parallelize the Jacobi algorithm for a 2D grid. (10)
(b) Explain the difference between `critical` and `atomic` sections. (4)

OR

16. (a) Explain virtual topologies in MPI and the use of `MPI_Cart_create()`. (7)
(b) Prove that parallel efficiency decreases as the number of processors increases for a fixed problem size (Amdahl's Law). (7)
17. (a) Discuss the major performance issues in MPI programs, namely serialization, contention, and message aggregation. (7)
(b) Describe collective communication operations in MPI—Broadcast, Scatter, Gather, and Reduction. (7)

OR

18. (a) Explain the MPI programming model with respect to process management and point-to-point communication. (7)
(b) Clearly distinguish between blocking and non-blocking communication using suitable examples. (7)
19. (a) Define synchronization, serialization, and contention in MPI programs. Explain how each affects parallel efficiency. (7)
(b) Why does excessive synchronization degrade MPI performance? (7)

OR

20. (a) Compare blocking, nonblocking, and asynchronous communication in MPI. (7)
(b) Explain techniques to reduce communication overhead in MPI applications. (7)

B24CS4P34	MODERN DATABASES	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course provides an exposure to the concepts and techniques in modern data management. Different types of NoSQL databases, their architecture and use cases are discussed in this course. A better understanding of data management is provided through MongoDB, Cassandra and Neo4J databases. This course helps the learners to develop applications that manage data efficiently with the help of suitable data models and techniques.

Prerequisites

Basic knowledge in DBMS

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Analyze issues and solutions in modern databases. (Cognitive Knowledge Level: Understand)
CO 2	To explore internal design, consistency, and distributed storage mechanisms. (Cognitive Knowledge Level: Understand)
CO 3	Employ and operate on document databases and techniques. (Cognitive Knowledge Level: Apply)
CO 4	Research, analyze and use emerging technologies in column store. (Cognitive Knowledge Level: Apply)
CO 5	Practice techniques in Graph database. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2		1		3			1		
CO 2	3	2	1								
CO 3	1	1	2		3			3	1	3	
CO 4	1	2		3	2	3	3		2		3
CO 5	2	2	2	1	3			3	1	3	2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Foundations of Modern and NoSQL Databases

Motivation for NoSQL; limitations of RDBMS, NoSQL database types – key-value, document, column-family, graph, ACID vs BASE, CAP Theorem, Consistency Models (strong, eventual, causal), Fragmentation, Replication, Sharding, and Scaling strategies.

MODULE 2 (7 hours)

Internal Design and System Architecture of Modern Databases

Key-Value stores – architecture, consistent hashing, lookup mechanisms, Internal storage: LSM Trees, memtables, WAL (Write-Ahead Log), Concurrency and recovery mechanisms, Replication consistency maintenance (quorum, hinted handoff), Indexing, caching in distributed databases.

MODULE 3 (7 hours)

Document Databases and JSON Data Model (MongoDB)

Document model and BSON/JSON structure, CRUD operations, Indexes, Aggregation framework, Data modeling (embedding vs referencing), Sharding and replication mechanisms, sharding architecture- consistency and locking, read and write operations.

MODULE 4 (7 hours)

Column-Family Databases and Storage Structures (Cassandra / HBase)

Architecture of Cassandra / HBase, Data model: keyspace, column families, clustering keys, Memtables, Write path, compaction and tunable consistency, Replication, consistency levels, and gossip protocol, Querying using CQL, Scalability and fault tolerance.

MODULE 5 (7 hours)

Graph Databases and Graph Modeling (Neo4j)

Graph data model – nodes, relationships, properties, Property graphs vs RDF triple stores, Data modelling principles and normalization of relationships, Cypher query language – patterns, filters, paths, Basic graph algorithms: traversal, shortest path, centrality, recommendations (introductory level only), Indexes, constraints, and query optimization

TextBooks

1. Pramod J. Sadalage, P. J., & Martin Fowler, M. (2012). NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. Boston, MA: Addison-Wesley.
2. Shashank Tiwari, S. (2011). Professional NoSQL. Indianapolis, IN: Wiley.

Reference Books

1. Kristina Chodorow. (2013). MongoDB: The Definitive Guide. Sebastopol, CA: O'Reilly Media.
2. Ian Robinson, Jim Webber, Emil Eifrem, Graph Databases (2nd Edition), O'Reilly.
3. Pramod J. Sadalage., & Martin Fowler. (2013). NoSQL Distilled. Boston, MA: Addison-Wesley.
4. Ian Robinson., Jim Webber., & Emil Eifrem. (2015). Graph Databases (2nd ed.). Sebastopol, CA: O'Reilly Media.
5. Jérôme Baton., & Rik Van Bruggen. (2017). Learning Neo4j 3.x (2nd ed.). Birmingham, UK: Packt Publishing.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture / Tutorial Hours
Total Hours		35 Hours
Module 1: Foundations of Modern and NoSQL Databases		7 hours
1.1	Motivation for NoSQL; limitations of RDBMS	1
1.2	NoSQL database types – key-value, document	1
1.3	Column-family, graph, ACID vs BASE	1
1.4	CAP Theorem	1
1.5	Consistency Models (strong, eventual, causal)	1
1.6	Fragmentation, Replication	1
1.7	Sharding and scaling strategies , Types of NoSQL Databases	1
Module 2: Internal Design and System Architecture of Modern Databases		7 hours
2.1	Key-Value stores – architecture	1
2.2	Consistent hashing, lookup mechanisms	1

2.3	Internal storage: LSM Trees	1
2.4	WAL (Write-Ahead Log)	1
2.5	Concurrency and recovery mechanisms	1
2.6	Consistency maintenance (quorum, hinted handoff)	1
2.7	Indexing, Caching in distributed databases	1
Module 3: Document Databases and JSON Data Model (MongoDB)		7 hours
3.1	Document model and BSON/JSON structure	1
3.2	CRUD operations	1
3.3	Indexes	1
3.4	Aggregation framework	1
3.5	Data modeling (embedding vs referencing)	1
3.6	Sharding and replication mechanisms	1
3.7	Sharding architecture – consistency and locking, read and write operations	1
Module 4: Column-Family Databases and Storage Structures (Cassandra/HBase)		7 hours
4.1	Architecture of Cassandra / HBase	1
4.2	Data model: keyspace, column families, clustering keys	1
4.3	Memtables, write path	1
4.4	Compaction and tunable consistency	1
4.5	Replication, consistency levels	1
4.6	Gossip protocol, Querying using CQL	1
4.7	Scalability and fault tolerance	1
Module 5: Graph Databases and Graph Modeling (Neo4j)		7 hours
5.1	Graph data model – nodes, relationships, properties	1
5.2	Property graphs vs RDF triple stores	1
5.3	Data modelling principles, Normalization of relationships	1
5.4	Cypher query language – patterns, filters, paths	1
5.5	Basic graph algorithms: traversal, shortest path, centrality, recommendations	1
5.6	Basic graph algorithms: traversal, shortest path, centrality, recommendations.	1
5.7	Indexes, constraints, and query optimization.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the limitations of relational databases in handling big data and high scalability requirements
2. Discuss the motivation for NoSQL databases and explain different types of NoSQL databases with suitable use cases

Course Outcome 2 (CO 2):

1. Explain the concept of CAP Theorem and discuss its impact on the design of distributed databases
2. Describe consistent hashing and explain how it supports scalability and fault tolerance in distributed systems

Course Outcome 3 (CO 3):

1. Explain CRUD operations in MongoDB with suitable command examples
2. Compare embedding and referencing techniques in MongoDB data modeling with examples

Course Outcome 4 (CO 4):

1. Analyze tunable consistency and replication strategies used in Cassandra
2. Explain the architecture of Cassandra and describe its write path in detail

Course Outcome 5 (CO 5):

1. Describe Cypher query language and explain how graph traversal queries are executed
2. Explain the graph data model used in Neo4j and compare property graphs with RDF triple stores

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CS4P34

Course Name: MODERN DATABASES

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Explain the limitations of traditional RDBMS that motivated the rise of NoSQL databases.
2. Briefly explain ACID vs BASE properties with any one example.
3. What is consistent hashing? Why is it required in distributed key-value stores?
4. Define Write-Ahead Log (WAL). What purpose does it serve?
5. Differentiate between embedding and referencing in MongoDB data modelling.
6. What is the role of indexes in MongoDB? Give an example.
7. Explain the concept of tunable consistency in Cassandra.
8. Define gossip protocol in Cassandra. Why is it essential?
9. Differentiate between property graphs and RDF triple stores.
10. What is a Cypher pattern? Give a simple example.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the CAP Theorem and discuss real-world scenarios representing CA, CP, and AP systems. (7)
(b) Describe fragmentation, replication and sharding with neat sketches. (7)

OR

12. (a) Compare key-value, document, column-family, and graph databases based on structure, use cases, and scalability. (7)
(b) Explain strong, eventual and causal consistency models with examples. (7)
13. (a) Describe Write-Ahead Logging (WAL) and its role in recovery. (7)
(b) Explain indexing and caching mechanisms in distributed databases. (7)

OR

14. (a) Describe the architecture of key-value stores with a neat diagram. (7)
(b) Explain the roles of memtables and LSM trees in modern storage engines. (7)
15. (a) Explain quorum-based replication and hinted handoff with suitable examples. (7)
(b) A MongoDB database contains customer documents with nested order details. Write queries to:
i. Retrieve customers with orders above a certain value
ii. Update order status for a specific customer (7)

OR

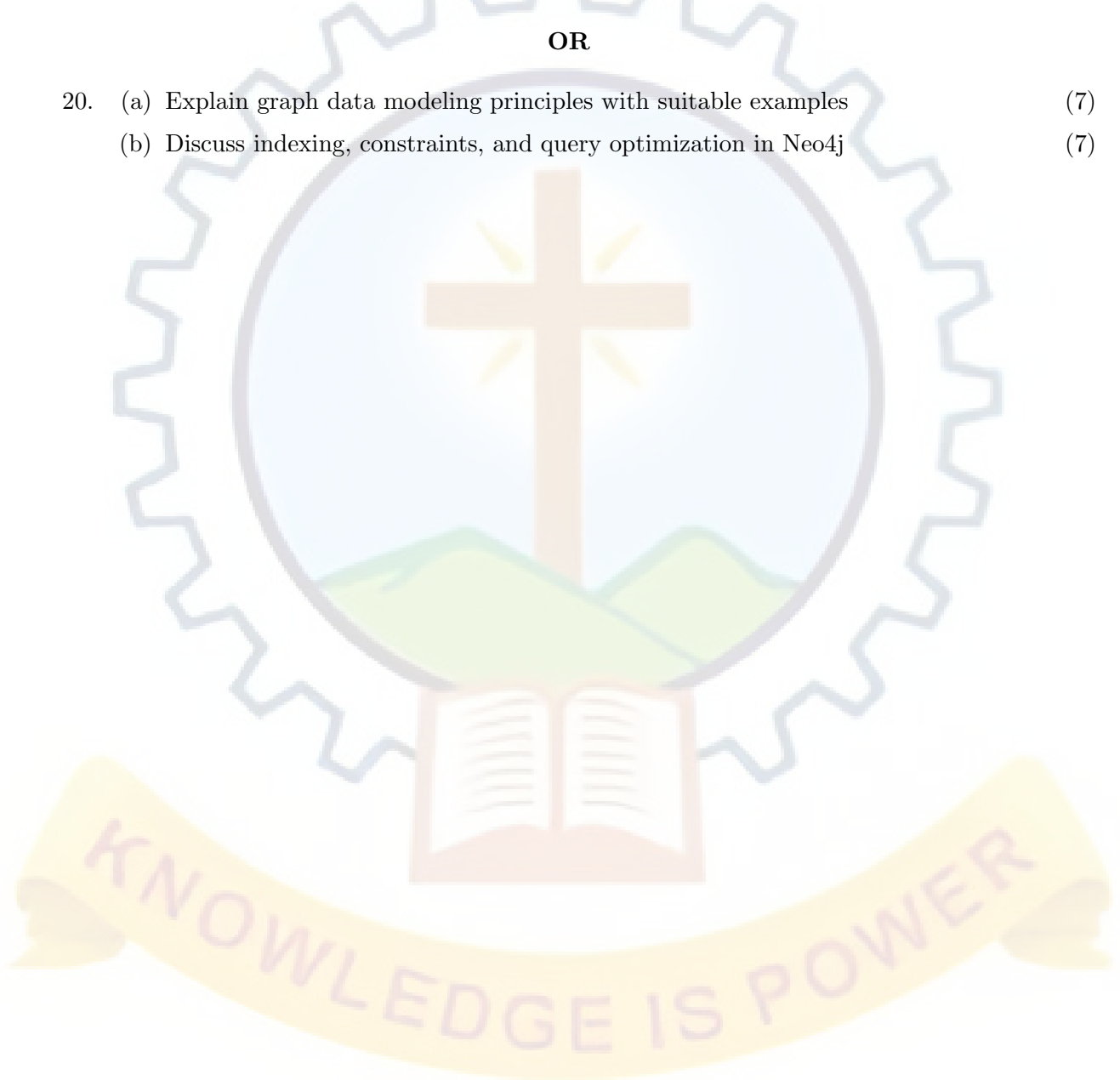
16. (a) A library stores data about books in a books collection:
{ "isbn": "978-1234567890", "title": "Data Systems", "author": "Tom Rick", "copies": 5, "categories": ["Databases", "Technology"] }
i. Insert a new book into the collection.
ii. Update the number of copies of a book when 2 books are issued.
iii. Retrieve the details of all books written by "Tom Rick". (8)
(b) Explain MongoDB sharding architecture and its benefits (6)
17. (a) A Cassandra table stores product sales data with the fields productid, storeid, saledate, and quantity.
i. Identify a suitable partition key and clustering key for the table.
ii. Write CQL commands to insert a record and retrieve sales details of a product from a specific store (7)
(b) Explain Cassandra write and read path in detail. (7)

OR

18. (a) Discuss scalability and fault tolerance mechanisms in Cassandra (7)
(b) Describe compaction strategies in Cassandra (7)
19. (a) Explain graph data modeling principles with suitable examples. (7)
(b) A social networking application stores users and their connections using a graph database.
i. Identify the nodes, relationships, and properties in this graph model.
ii. Write a Cypher query to find all friends of a given user. (7)

OR

20. (a) Explain graph data modeling principles with suitable examples (7)
(b) Discuss indexing, constraints, and query optimization in Neo4j (7)



B24CS4P41	SOFTWARE TESTING ARCHITECTURE	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4		2024

Preamble

This is a course in theoretical computer science that introduces the concepts and methods in software testing. It covers various techniques for test case design used to test software artifacts, including requirements, design, and code, the different techniques for test case design based on graphs, programming language syntaxes and symbolic execution using PEX tool. It enables the learners to follow a systematic software testing approaches while developing applications.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	List a range of different software testing techniques and be able to apply specific unit testing method to the projects using Junit.(Cognitive Knowledge Level: Understand)
CO 2	Illustrate using appropriate tools the mutation testing method for a given piece of code to identify hidden defects that can't be detected using other testing methods.(Cognitive Knowledge Level: Apply)
CO 3	Explain graph coverage criteria in terms of control flow graph and data flow graph for a given program.(Cognitive Knowledge Level: Understand)
CO 4	Demonstrate the importance of black-box approaches in terms of domain and functional testing.(Cognitive Knowledge Level: Apply)
CO 5	Illustrate the use of PEX tool with symbolic execution.(Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1		1	2		1	1			1
CO 2	2	1	1	2	1	1		1			
CO 3	2	1		1	1						
CO 4	2	1	1	2	1						
CO 5	2	1	1	2	2						1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(Quiz/Seminar/microproject/video assignments/Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction to Software Testing

Some Popular Errors – Ariane 5, Therac 25, Intel Pentium Bug. What is Software testing? Why should it be tested? Software Quality, Role of Testing. Testing Process - Level 0 thinking, Level 1 thinking, Level 2 thinking, Level 3 thinking, Level 4 thinking. Software Testing Terminologies - Verification, Validation and Testing, Faults, Error and Bug, Test cases, Coverage Criteria. Types of Testing- Unit testing, integration testing, System testing, Acceptance testing, Beta testing, Functional testing, Stress testing, Performance testing, Usability testing and Regression testing. Testing Methods - Black Box testing, White Box testing, Grey Box testing.

MODULE 2 (8 hours)

Unit Testing

Concept of Unit testing. Static Unit testing. Dynamic Unit testing - Control Flow testing, Data Flow testing, Domain testing, Functional Program testing. Mutation testing - Mutation and Mutants, Mutation operators, Mutation score. Junit - Framework for Unit testing. Case Study - Mutation testing using Junit and Muclipse.

MODULE 3 (8 hours)

Unit Testing - White Box Approaches

Overview of Graph Coverage Criteria. Structural Graph Coverage Criteria - Node/vertex coverage, Edge coverage, Edge pair coverage, Path coverage, Complete path coverage, Prime path coverage, Complete round trip coverage, Simple round trip coverage. Data Flow Criteria - du paths, du pairs. Subsumption Relationships among Graph Coverage Criteria. Graph Coverage for Source Code - Control flow graphs for code, CFG: If statement, CFG: If statement with return, CFG: Switch-case, CFG: Loops, CFG: Exceptions (try-catch). Example program – Statistics. Graph Coverage for Design Elements - Call graphs and classes, Class inheritance testing: Coverage criteria, Coverage criteria on inheritance graph, Data flow at the design level, Inter-procedural DU pairs, Coupling du-pairs example. Example - Quadratic Root. Case Study - Graph Based testing using JUnit Framework.

MODULE 4 (8 hours)

Unit Testing - Black Box Approaches

Domain Testing / Input Space Partitioning - Partitions of a set. Input domain modelling - Interface-based approach, Functionality-based approach. Identifying values. Multiple partitions of the input domain - All Combinations Coverage (ACoC), Each Choice Coverage (ECC), Pair-wise Coverage, T-wise Coverage, Base Choice Coverage, Multiple Base Choices Coverage. TriTyp example. Functional Testing - Functional Testing Concepts of Howden. Functional testing - Important Steps. Types of Functional testing - Equivalence Class Partitioning, Boundary Value Analysis, Decision Tables, Random Testing. Case Study - Black Box testing approaches using JUnit.

MODULE 5 (5 hours)

Grey Box Testing Approaches

Introduction to Grey Box testing - Why Grey Box testing, Gray Box Methodology, Advantages and Disadvantages. Techniques of Grey Box Testing - Matrix Testing, Regression Testing, Orthogonal Array Testing or OAT, Pattern Testing. An Introduction to PEX - Parameterized Unit Testing, The Testing Problem. Symbolic Execution – Example, Symbolic execution tree. PEX application Case Study – PEX.

TextBooks

1. Paul Ammann and Jeff Offutt, Introduction to Software Testing, Cambridge University Press 2/e, 2016
2. Kshirasagar Naik and Priyadarshi Tripathy, Software Testing And Quality Assurance: Theory And Practice, Wiley 1/e, 2008.

Reference Books

1. King, James C, “Symbolic Execution and Program Testing”, Association for Computing Machinery, July 1976.
2. Ron Patten, “Software Testing”, Pearson 2/e, 2005.
3. Paul C. Jorgensen, “Software Testing: A Craftsman’s Approach “, CRC Press 4/e, 2017
4. Dorothy Graham, Rex Black, Erik van Veenendaal, “Foundations of Software Testing “, Cengage 4/e, 2021

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to Software Testing		7 hours
1.1	Some Popular Errors – Ariane 5, Therac 25, Intel Pentium Bug. What is Software testing? Why should it be tested?	1
1.2	Software Quality, Role of Testing. Testing Process - Level 0 thinking, Level 1 thinking, Level 2 thinking, Level 3 thinking, Level 4 thinking	1
1.3	Software Testing Terminologies - Verification, Validation and Testing, Faults, Error and Bug, Test cases, Coverage Criteria.	1
1.4	Types of Testing- Unit testing, integration testing, System testing, Acceptance testing	1

1.5	Beta testing, Functional testing, Stress testing	1
1.6	Performance testing, Usability testing and Regression testing.	1
1.7	Testing Methods - Black Box testing, White Box testing, Grey Box testing	1
Module 2: Unit Testing		8 hours
2.1	Concept of Unit testing. Static Unit testing.	1
2.2	Dynamic Unit testing - Control Flow testing,	1
2.3	Data Flow testing, Domain testing,	1
2.4	Functional Program testing .	1
2.5	Mutation testing - Mutation and Mutants,	1
2.6	Mutation operators, Mutation score	1
2.7	Junit - Framework for Unit testing.	1
2.8	Case Study - Mutation testing using JUnit and Muclipse	1
Module 3: Unit Testing - White Box Approaches		8 hours
3.1	Overview of Graph Coverage Criteria. Structural Graph Coverage Criteria - Node/vertex coverage, Edge coverage, Edge pair coverage	1
3.2	Path coverage, Complete path coverage, Prime path coverage, Complete round trip coverage, Simple round trip coverage.	1
3.3	Data Flow Criteria - du paths, du pairs. Subsumption Relationships among Graph Coverage Criteria	1
3.4	Graph Coverage for Source Code - Control flow graphs for code, CFG: If statement, CFG: If statement with return,	1
3.5	CFG: Switch-case, CFG: Loops, CFG: Exceptions (try-catch). Example program – Statistics.	1
3.6	Graph Coverage for Design Elements - Call graphs and classes, Class inheritance testing: Coverage criteria, Coverage criteria on inheritance graph,	1
3.7	Data flow at the design level, Inter-procedural DU pairs, Coupling du-pairs example. Example - Quadratic Root.	1
3.8	Case Study - Graph Based testing using JUnit Framework.	1
Module 4: – Unit Testing - Black Box Approaches		8 hours
4.1	Domain Testing / Input Space Partitioning - Partitions of a set. Input domain modelling	1
4.2	Interface-based approach, Functionality-based approach	1
4.3	Identifying values. Multiple partitions of the input domain - All Combinations Coverage (ACoC), E	1

4.4	Each Choice Coverage (ECC), Pair-wise Coverage, T-wise Coverage, Base Choice Coverage, Multiple Base Choices Coverage	1
4.5	TriTyp example. Functional Testing - Functional Testing Concepts of Howden. Functional testing - Important Steps.	1
4.6	Types of Functional testing - Equivalence Class Partitioning,	1
4.7	Boundary Value Analysis, Decision Tables, Random Testing.	1
4.8	Case Study - Black Box testing approaches using JUnit.	1
Module 5: Grey Box Testing Approaches		5 hours
5.1	Introduction to Grey Box testing - Why Grey Box testing, Gray Box Methodology,	1
5.2	Advantages and Disadvantages. Techniques of Grey Box Testing - Matrix Testing,	1
5.3	Regression Testing, Orthogonal Array Testing or OAT, Pattern Testing.	1
5.4	An Introduction to PEX - Parameterized Unit Testing, The Testing Problem.	1
5.5	Symbolic Execution – Example, Symbolic execution tree. PEX application Case Study – PEX.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the following types of testing methods with examples.
 - (a) Black-box testing.
 - (b) White-box testing.
 - (c) Grey-box testing.

Course Outcome 2 (CO 2):

```
public static int power(int left , int right)
{
    //*****
    // Raises left to the power of right
    // precondition : right >= 0
    // postcondition: Returns left**right
    //*****
}
```

```
int rslt;  
rslt = left;  
  
if (right == 0)  
{  
    rslt = 1;  
}  
else  
{  
    for (int i = 2; i <= right; i++)  
        rslt = rslt * left;  
}  
  
return rslt;  
}
```

Course Outcome 3 (CO 3):

1. Draw the control flow graph and data flow graph of given piece of code.

```
public static double ReturnAverage(int value[], int AS, int MIN,  
int MAX) {
```

```
/*
```

```
Function: ReturnAverage
```

```
Computes the average of all those numbers in the input array in  
the positive range [MIN, MAX]. The maximum size of the array is AS.  
But, the array size could be smaller than AS in which case the end  
of input is represented by -999.
```

```
*/
```

```
int i, ti, tv, sum;  
double av;
```

```
i = 0;  
ti = 0;  
tv = 0;  
sum = 0;
```

```
while (ti < AS && value[i] != -999) {  
    ti++;  
    if (value[i] >= MIN && value[i] <= MAX) {  
        tv++;  
        sum = sum + value[i];  
    }  
    i++;  
}
```

```
if (tv > 0)  
    av = (double) sum / tv;  
else
```

```
    av = (double) -999;

    return av;
}
```

Course Outcome 4 (CO 4):

1. Explain the following with examples.
 1. Input domain modelling.
 2. All Combinations Coverage (ACoC)
 3. Each Choice Coverage (ECC)
 4. Pair-wise Coverage
 5. T-wise Coverage
 6. Base Choice Coverage
 7. Multiple Base Choices Coverage

Course Outcome 5 (CO 5):

1. Draw the symbolic execution tree for the following program code and explain the symbolic execution of testme (a1, a2).

```
int twice (int v) {
    return 2 * v;
}

void testme (int x, int y) {
    z = twice(y);
    if (z == x) {
        if (x > y + 10)
            ERROR;
    }
}

int main() {
    x = sym_input();
    y = sym_input();
    testme(x, y);
    return 0;
}
```


MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CS4P41

Course Name: SOFTWARE TESTING ARCHITECTURE

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Explain the differences between Validation and Verification?
2. Explain the differences between Fault, Error, and Bug?
3. Define Ground string, Mutation score, and Mutants?
4. What are the functions of Test driver and Test stubs in dynamic unit testing?
5. Define Node coverage, Edge coverage and Prime path coverage in a control flow graph?
6. What are du paths and du pairs in a data flow graph?
7. Explain the two approaches in input domain modelling?
8. Explain the difference between Equivalence Class Partitioning and Boundary Value Analysis?
9. Briefly explain three techniques of Grey box testing?
10. Explain the concept of symbolic execution with the help of a toy example?

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the following types of testing (i) Black Box testing (ii) White Box testing (iii) GreyBox testing (iv) Unit testing (v) Integration testing (vi) System testing (vii) Acceptance testing (14)

OR

12. (a) Explain the following coverage criterias based on the code fragment given below? (i) Functional coverage (ii) Statement coverage (iii) Conditional coverage (iv) Branch coverage

```
int for (int x, int y){  
    int z = 0;  
    if ((x > 0) && (y > 0)){  
        z = x;  
    }  
    return z;  
}
```

(8)

- (b) Write positive and negative test cases for an ATM Machine? (6)

13. (a) Explain Dynamic unit test environment with a neat figure. (8)

- (b) Explain the major difference between control flow testing and data flow testing. (6)

OR

14. (a) Explain seven types of mutation operators with neat examples? (14)

15. (a) Explain touring, side trips and detours with a neat example (7)

- (b) Why does prime path coverage subsume edge and node coverage? Give justification with an example graph. (7)

OR

16. (a) Draw CFG fragment for (i) Simple if (ii) Simple while loop (iii) Simple for loop (7)

- (b) Explain the following concepts with examples? (i) Call graph (ii) Inheritance graph (iii) Coupling du-pairs (7)

17. (a) What are the four important steps in functional testing? (7)

- (b) Briefly explain input domain modelling approaches? (7)

OR

18. (a) Consider the triangle classification program with a specification: The program reads floating values from the standard input. The three values A, B, and C are interpreted as representing the lengths of the sides of triangle. The program then prints a message to the standard output that states whether the triangle, if it can be formed, is scalene, isosceles, equilateral, or right angled. Determine the following for the above program:

(i) Boundary Condition $A + B > C$ (Scalene Triangle)

, identify test cases to verify the boundary.

(ii) Boundary Condition $A = C$ (Isosceles Triangle)

, identify testcases to verify the boundary.

(ii) Boundary Condition $A = B = C$ (Equilateral Triangle)

, identify testcases to verify the boundary.

(6)

(b) Develop a decision table to generate test cases for this specification.

(8)

19. (a) Explain the importance of grey box testing, its advantages and disadvantages?

(9)

(b) Explain the concept of symbolic execution tree?

(5)

OR

20. (a) Consider the code fragment given below:

```
1. POWER: PROCEDURE(X, Y);  
2. Z ← 1;  
3. J ← 1;  
4. LAB: IF Y  $\geq$  J THEN  
5.     DO; Z ← Z * X;  
6.     J ← J + 1;  
7.     GO TO LAB; END;  
8. RETURN (Z);  
9. END;
```

Explain Symbolic execution of POWER (α_1, α_2).

(7)

(b) Explain Execution tree for POWER (α_1, α_2).

(7)

B24CS4P42	CYBER FORENSICS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

The course on Cyber Forensics aims at exploring the basics of Cyber Forensics and Cyber security, the forensic investigation process and principles and the different types of cybercrimes and threats. This course also focuses on the forensic analysis of File systems, the Network, the Windows and Linux Operating systems. The course gives a basic understanding of the forensics analysis tools and a deep understanding of Anti forensics practices and methods. All the above aspects are dealt with case studies of the respective areas.

Prerequisites

Knowledge in File Systems, Operating systems, Networks and a general awareness on Cyber Technologies.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the basic concepts in Cyber Forensics, Forensics Investigation Process and Cyber security (Cognitive Knowledge Level: Understand)
CO 2	Infer the basic concepts of File Systems and its associated attribute definitions (Cognitive Knowledge Level: Understand)
CO 3	Utilize the methodologies used in data analysis and memory analysis for detection of artefacts (Cognitive Knowledge Level: Apply)
CO 4	Identify web attacks and detect artefacts using OWASP and penetration testing. (Cognitive Knowledge Level: Apply)
CO 5	Summarize anti-forensics practices and data hiding methods. (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	1	1	1	2	3	3	3		2
CO 2	2	2	2	2	2	1		1	1	1	2
CO 3	3	3	3	3	3	2	2	3	3	3	3
CO 4	2	3	3	3	3	3	3	3	3	3	3
CO 5	2	2	2	3	3	3	3	3	3	2	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/ Seminar/ Microproject/ Video Assignments/ Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (8 hours)

Cyber Forensics and Cyber Security

Computer Forensics: History of computer forensics, preparing for computer investigations, understanding Public and private investigations- Forensics Investigation Principles - Forensic Protocol for Evidence Acquisition - Digital Forensics -Standards and Guidelines - Digital Evidence – Data Acquisition - storage formats for digital evidence, determining the best acquisition method, contingency planning for image acquisitions, Cyber Forensics tools- Challenges in Cyber Forensics, Skills Required to Become a Cyber Forensic Expert Cyber Security: Cybercrimes, Types of Cybercrimes - Recent Data Breaches - Recent Cyber security Trends - Case Study: Sim Swapping Fraud, ATM Card Cloning, Hacking email for money, Google Nest Guard, Email Crimes, Phishing, Types of Phishing.

MODULE 2 (7 hours)

File System Forensics

File system Analysis: FAT and NTFS concepts and analysis -File system category, Content category, Metadata category, File name category, Application category,Application-level search techniques, Specific file systems, File recovery, Consistency check. FAT data structure-Boot sector, FAT 32 FS info, directory entries, Long file name directory entries

MODULE 3 (8 hours)

Operating System Forensics

Windows Forensics: Live Response- Data Collection- Locard's Exchange Principle, Order of Volatility Volatile and Non Volatile Data Live-Response Methodologies: Data Analysis- Agile Analysis, Windows Memory Analysis, Rootkits and Rootkit detection. Linux Forensics: Live Response Data Collection- Prepare the Target Media, Format the Drive, Gather Volatile Information, Acquiring the Image, Initial Triage, Data Analysis- Log Analysis, Keyword Searches, User Activity, Network Connections, Running Processes, Open File Handlers, The Hacking Top Ten, Reconnaissance Tools

MODULE 4 (7 hours)

Network Forensics

The OSI Model, Forensic Footprints, Seizure of Networking Devices, Network Forensic Artifacts, ICMP Attacks, Drive-By Downloads, Network Forensic Analysis Tools, Case Study: Wireshark. Web Attack Forensics: OWASP Top 10, Web Attack Tests, Penetration Testing.

MODULE 5 (6 hours)

Anti-Forensics

Anti-forensic Practices - Data Wiping and Shredding- Data Remanence, Degaussing, Case Study: USB Oblivion, Eraser - Trail Obfuscation: Spoofing, Data Modification, Case Study: Timestamp – Encryption, Case Study: VeraCrypt, Data Hiding: Steganography and Cryptography, Case Study: SilentEye, Anti-forensics Detection Techniques, Case Study: Stegdetect

Text Books

1. Bill Nelson, Amelia Phillips and Christopher Steuart, Computer forensics - Guide to Computer Forensics and Investigations, 4/e, Course Technology Inc.
2. Brian Carrier, File System Forensic Analysis, Addison Wesley, 2005.
3. Harlan Carvey, Windows Forensic Analysis DVD Toolkit, 2/e, Syngress
4. Cory Altheide, Todd Haverkos, Chris Pogue, Unix and Linux Forensic Analysis DVD Toolkit, 1/e, Syngress.
5. William Stallings, Network Security Essentials Applications and Standards, 4/e, Prentice Hall
6. Eric Maiwald, Fundamentals of Network Security, McGraw-Hill, 2004.

Reference Books

1. Michael. E. Whitman, Herbert. J. Mattord, Principles of Information Security, Course Technology, 2011.
2. William Stallings, Cryptography and Network Security Principles and Practice, 4/e, Prentice Hall.
3. Niranjana Reddy, Practical Cyber Forensics: An Incident-Based Approach to Forensic Investigations, Apress, 2019.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Cyber Forensics and Cyber Security		8 hours
1.1	History of computer forensics, preparing for computer investigations	1
1.2	Understanding Public and Private Investigations- Forensics Investigation Principles	1

1.3	Forensic Protocol for Evidence Acquisition, Digital Forensics -Standards and Guidelines - Digital Evidence	1
1.4	Data Acquisition - storage formats for digital evidence, determining the best acquisition method	1
1.5	Contingency planning for image acquisitions, Cyber Forensics tools	1
1.6	Challenges in Cyber Forensics, Skills Required to Become a Cyber Forensic Expert Cybercrimes, Types of Cybercrimes - Recent Data Breaches - Recent Cyber security Trends	1
1.7	Case Study: Sim Swapping Fraud, ATM Card Cloning, Case Study: Hacking email for money, Google Nest Guard	1
1.8	Email Crimes, Phishing, Types of Phishing	1
Module 2:File System Forensics		7 hours
2.1	FAT and NTFS concepts and analysis	1
2.2	File system category, Content category	1
2.3	Metadata category	1
2.4	File name category, Application category, Application-level search techniques	1
2.5	Specific file systems, File recovery, Consistency check	1
2.6	FAT data structure-Boot sector	1
2.7	FAT 32 FS info, directory entries, Long file name directory entries	1
Module 3: Operating System Forensics		8 hours
3.1	Live Response- Data Collection- Locard's Exchange Principle	1
3.2	Order of Volatility, Volatile and Non-Volatile Data	1
3.3	Live-Response Methodologies: Data Analysis- Agile Analysis	1
3.4	Windows Memory Analysis, Rootkits and Rootkit detection	1
3.5	Linux Forensics: Live Response Data Collection	1
3.6	Prepare the Target Media, Format the Drive, Gather Volatile Information, Acquiring the Image, Initial Triage	1
3.7	Data Analysis- Log Analysis, Keyword Searches, User Activity, Data Analysis- Network Connections, Running Processes, Open File Handlers	1
3.8	The Hacking Top Ten, Reconnaissance Tools	1
Module 4: Network Forensics		7 hours
4.1	OSI Model	1

4.2	Forensic Footprints, Seizure of Networking Devices, Network Forensic Artifacts	1
4.3	ICMP Attacks, Drive-By Downloads, Network Forensic Analysis Tools	1
4.4	Web Attack Forensics	1
4.5	OWASP Top 10, Web Attack Tests	1
4.6	Penetration Testing-1	1
4.7	Penetration Testing.-2	1
Module 5: Anti-Forensics		6 hours
5.1	Anti-forensic Practices - Data Wiping and Shredding	1
5.2	Data Remanence, Degaussing	1
5.3	Trail Obfuscation: Spoofing, Data Modification	1
5.4	Role of Encryption in Forensics	1
5.5	Data Hiding: Steganography and Cryptography	1
5.6	Anti-forensics Detection Techniques	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. Explain the Forensics principles and protocols for evidence acquisition. Discuss the different cyber forensics tools used for image acquisition.
2. Explain various storage formats for digital evidence and their significance in investigations.

Course Outcome 2 (CO 2):

1. Explain the pros and cons of NTFS and FAT File systems. Also give the challenges the investigators would face in extracting evidences from these file systems.
2. Illustrate the structure and working of the FAT file system, including the boot sector, FAT table, and directory entries.

Course Outcome 3 (CO 3):

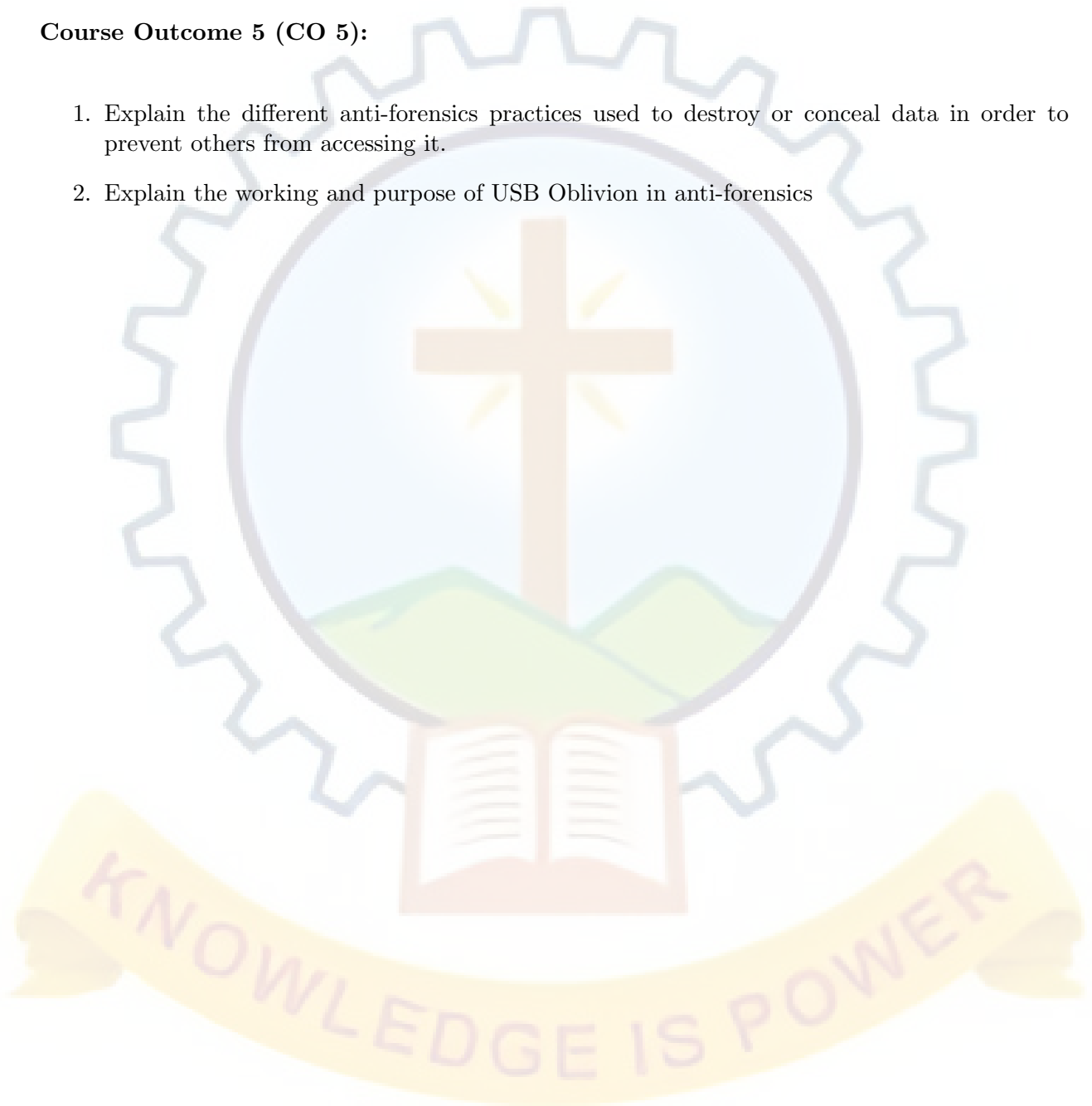
1. Apply any memory forensics methodologies/tools to extract volatile and nonvolatile data from a Windows based system.
2. Describe Locard's Exchange Principle and explain its relevance in digital investigations

Course Outcome 4 (CO 4):

1. Use web attacks test tools like netcraft to identify web application vulnerabilities of a particular site say www.xyz.com
2. Explain ICMP attacks and discuss how they can be detected using forensic techniques

Course Outcome 5 (CO 5):

1. Explain the different anti-forensics practices used to destroy or conceal data in order to prevent others from accessing it.
2. Explain the working and purpose of USB Oblivion in anti-forensics



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CS4P42

Course Name: CYBER FORENSICS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Distinguish between public and private investigations.
2. What are the three computer forensics data acquisitions formats?
3. List any three features of NTFS which are not in FAT.
4. Define the terms file slack, RAM slack and drive slack.
5. What is Locard's exchange principle? Why is it important in forensic investigations?
6. Why would you conduct a live response on a running system?
7. What are the different tools used in Network Forensics?
8. Explain how Risk Analysis and Penetration Testing are different.
9. Why we are using Steganography?
10. How is data wiping done in hard drive?

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Discuss the different types of Cybercrimes. List the tools used for identifying Cyber Crimes. (8)
(b) Differentiate between Static acquisition and Live acquisition with example. (6)

OR

12. (a) Explain the principles of Digital Forensic Investigation? Why is it important? Comment. (7)
(b) When you perform an acquisition at a remote location, what should you consider preparing this task? (7)
13. (a) Discuss the FAT File Structure. (8)
(b) Does Windows NT use FAT or NTFS? Explain. (6)

OR

14. (a) What is Metadata? Discuss the first 16 metadata records you would find in the MFT? (7)
(b) Explain the different data categories in a File System. (7)
15. (a) What is Agile requirement analysis? (8)
(b) Explain the different types of volatile information in a live response system. List any two tools used for obtaining volatile information. (6)

OR

16. (a) What are the main live response methodologies? (7)
(b) What is Physical Memory Dump? Explain how a physical memory dump is analysed. (7)
17. (a) What is OWASP? Also mention the Top 10 web application vulnerabilities in 2021. (8)
(b) How would you setup Wireshark to monitor packets passing through an internet router? (6)

OR

18. (a) What are the goals of conducting a pentesting exercise? (7)
(b) Discuss the types of penetration testing methodologies. (7)
19. (a) How is Steganography done? (8)
(b) Why does data need Cryptography? (6)

OR

20. (a) Explain the different types of Anti-forensics Detection Techniques. (7)
(b) What is Spoofing? How to prevent spoofing attack? (7)

B24CS4P43	EMBEDDED SYSTEMS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

The objective of this course is to familiarize learners with the technologies behind embedded computing systems. This course provides foundational knowledge of embedded system architecture, 8051-based design, hardware–software co-design principles, and firmware development using Embedded C. It aims to prepare students for real-world embedded system design and implementation.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the core components, characteristics, and applications of embedded systems, and their difference from general computing systems. (Cognitive Knowledge Level: Understand)
CO 2	Apply knowledge of the 8051 microcontroller, its architecture, instruction set, and addressing modes, to design and develop embedded systems. (Cognitive Knowledge Level: Apply)
CO 3	Develop embedded firmware using appropriate languages and understand the key concepts in hardware–software co-design. (Cognitive Knowledge Level: Apply)
CO 4	Use the integration of embedded hardware and firmware, and utilize tools for system testing and validation. (Cognitive Knowledge Level: Apply)
CO 5	Apply and explain the application of embedded systems in diverse domains such as mobile communication, automotive electronics, RFID, wireless sensor networks, robotics, biomedical systems, and brain–machine interfaces. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	2	1								
CO 2	3	3	3		2						
CO 3	3	3	3	2	3				1		
CO 4	2	2	3	3	3				1		
CO 5	3	2	1			1	1				

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study (quiz / seminar / micro project / video assignments / analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Introduction to Embedded Systems:

Definition of Embedded System, Embedded Systems Vs General Computing Systems, History, Classification, and Major application areas of Embedded Systems, Purpose of Embedded Systems; Typical system - Core of the Embedded System, Memory, Sensors and Actuators, Communication Interface, Embedded Firmware, Other System components; Characteristics and Quality attributes of Embedded Systems.

MODULE 2 (8 hours)

Designing with 8051:

Factors to be Considered in Selecting a Controller, Why 8051bMicrocontroller, Designing with 8051, The 8052 Microcontroller, 8051/52 Variants; Different Addressing Modes Supported by 8051; The 8051 Instruction Set; Fundamental Issues in Hardware Software Co-Design; Computational Models in Embedded Design; Introduction to Unified Modelling Language (UML); Hardware Software Trade-offs.

MODULE 3 (8 hours)

Design and Development:

Hardware Design and Development - VLSI and Integrated Circuit Design, Electronic Design Automation (EDA) Tools, The PCB Layout Design, Printed Circuit Board (PCB) Fabrication; Firmware Design and Development - Embedded Firmware Design, Embedded Firmware Development Languages, Programming in Embedded C.

MODULE 4 (7 hours)

Integration and Testing of Embedded Hardware and Firmware:

Integration of Hardware and Firmware, Boards Bring up, The Embedded System Development Environment - The Integrated Development Environment (IDE), Types of files generated on CrossCompilation, Disassembler/Decompiler, Simulators, Emulators and Debugging, Target Hardware Debugging, Boundary Scan.

MODULE 5 (7 hours)

Emerging Application Domains:

Mobile phone, automotive electronics, radio frequency identification (RFID), wireless sensor networks(WISENET), robotics, biomedical applications, brain machine interface. Case Study : Battery-Operated Smartcard Reader, Automated Meter Reading System (AMR), Digital Camera

Text Books

1. K. V. Shibu, Introduction to Embedded Systems, McGraw Hill Education, Second Edition, 2017.
2. Lyla B Das, Embedded systems: An Integrated Approach, 1st Ed., Pearson, 2013

Reference Books

1. Raj Kamal, Embedded Systems: Architecture, Programming and Design. Tata McGraw-Hill Education, 2011.
2. James K. Peckol, Embedded Systems: A Contemporary Design Tool, John Wiley & Sons, Second Edition, 2019.
3. Marilyn Wolf, Computers as Components-Principles of Embedded Computing System Design, Morgan Kaufmann, Elsevier, Fourth Edition, 2016.
4. Frank Vahid & Tony Givargis, Embedded System Design, 2nd Edition, John Wiley.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to Embedded Systems		6 hours
1.1	Definition of Embedded System, Embedded Systems Vs General Computing Systems	1
1.2	History, Classification, and, Major application areas of Embedded Systems	1
1.3	Purpose of Embedded Systems; Typical system - Core of the Embedded System	1
1.4	Memory, Sensors and Actuators, Communication Interface	1
1.5	Embedded Firmware, Other System components	1
1.6	Characteristics and Quality attributes of Embedded Systems	1
Module 2: Designing with 8051		8 hours
2.1	Factors to be Considered in Selecting a Controller, Why 8051 Microcontroller	1
2.2	Designing with 8051, The 8052 Microcontroller	1

2.3	8051/52 Variants; Different Addressing Modes Supported by 8051	1
2.4	The 8051 Instruction Set	1
2.5	Fundamental Issues in Hardware Software Co-Design	1
2.6	Computational Models in Embedded Design	1
2.7	Introduction to Unified Modelling Language (UML)	1
2.8	Hardware Software Trade-offs.	1
Module 3: Design and Development		8 hours
3.1	Hardware Design and Development	1
3.2	VLSI and Integrated Circuit Design	1
3.3	Electronic Design Automation (EDA) Tools	1
3.4	The PCB Layout Design	1
3.5	Printed Circuit Board (PCB) Fabrication	1
3.6	Firmware Design and Development - Embedded Firmware Design	1
3.7	Embedded Firmware Development Languages	1
3.8	TOPIC Programming in Embedded C.	1
Module 4: Integration and Testing of Embedded Hardware and Firmware		7 hours
4.1	Integration of Hardware and Firmware	1
4.2	Boards Bring up, The Embedded System Development Environment - IDE	1
4.3	Types of files generated on CrossCompilation	1
4.4	Disassembler/Decompiler, Simulators	1
4.5	Emulators and Debugging	1
4.6	Target Hardware Debugging	1
4.7	Boundary Scan	1
Module 5: Emerging Application Domains		7 hours
5.1	Mobile phone, automotive electronics	1
5.2	radio frequency identification (RFID)	1
5.3	wireless sensor networks(WISENET)	1
5.4	robotics, biomedical applications, brain machine interface.	2
5.5	Case Study : Battery-Operated Smartcard Reader	1
5.6	Automated Meter Reading System (AMR), Digital Camera	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Analyze the quality attributes of embedded systems and explain why reliability is critical in safety applications.
2. Illustrate a typical embedded system architecture and explain the role of each component.

Course Outcome 2 (CO 2):

1. Write an 8051 assembly program to:
Add two 8-bit numbers
Generate a square wave using a port pin
Transfer a block of data from one memory location to another
2. Design a simple traffic light controller using 8051. Provide algorithm and flowchart.

Course Outcome 3 (CO 3):

1. Design a PCB layout for a simple microcontroller-based LED interface system. Explain design considerations.
2. Develop an embedded C program to interface:
 - (a) LED and push button
 - (b) Temperature sensor and display
 - (c) Relay control system

Course Outcome 4 (CO 4):

1. Compare debugging using simulator, emulator, and real hardware debugging.
2. Develop a structured validation plan for testing a newly designed embedded controller board.

Course Outcome 5 (CO 5):

1. Describe the technical challenges involved in brain-machine interface systems.
2. Design a simple RFID-based access control system and explain its components.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CS4P43

Course Name: EMBEDDED SYSTEMS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. List any four characteristics of embedded systems.
2. What is meant by hard real-time system?
3. What are the main differences between 8051 and 8052?
4. What is the function of PSW register?
5. What is the purpose of PCB layout design?
6. What is hardware-software co-design?
7. List the types of files generated during cross-compilation.
8. What is target hardware debugging?
9. What is a brain-machine interface (BMI)?
10. State three applications of RFID technology.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the classification of embedded systems based on performance and functionality. (6)
(b) Explain how embedded systems are used in industrial automation with suitable examples. (8)

OR

12. (a) Compare and analyze an automobile ECU and a personal computer in terms of hardware, performance, and purpose. (8)
(b) Discuss the role of memory in embedded systems. (6)
13. (a) Develop an 8051-based system to interface an LED and switch. Explain hardware and software design. (10)
(b) Explain computational models used in embedded system design. (4)

OR

14. (a) Illustrate how different addressing modes are used in data transfer operations with examples. (8)
(b) Describe the importance of UML in embedded system modeling. (6)
15. (a) Illustrate the firmware development cycle with suitable example. (7)
(b) Explain how hardware constraints influence firmware design with an example. (7)

OR

16. (a) Design a PCB layout for a simple microcontroller-based LED interface system. Explain design considerations. (8)
(b) Describe the compilation process in embedded firmware development. (6)
17. (a) Illustrate the complete workflow from writing firmware code to flashing it onto target hardware. (8)
(b) Explain the working principle of boundary scan. (6)

OR

18. (a) Analyze the role of development tools in reducing embedded system design time. (10)
(b) Explain the role of IDE in embedded firmware development. (4)
19. (a) Develop a block diagram of a wireless sensor node and explain its operation. (7)
(b) Explain how embedded controllers are used in automotive airbag systems. (7)

OR

20. (a) Evaluate the importance of real-time processing in biomedical embedded applications. (7)
(b) Analyze how embedded systems enable autonomous behavior in robotics. (7)

B24CS4P44	CLIENT SERVER COMPUTING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

The course introduces the fundamental concepts and architecture of Client/Server computing, highlighting its role in modern distributed systems and business applications. It covers the design and functioning of client/server components, network services, database servers, and transaction processing mechanisms used in enterprise environments. The course also explores emerging web technologies, web services, and modern computing models that extend client/server architecture to support scalable and efficient applications.

Prerequisites

Basic knowledge in Computer Networks, Database Management Systems and Operating Systems

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the fundamental concepts, characteristics, and advantages of Client/Server computing and its role in modern computing environments. (Cognitive Knowledge Level: Understand)
CO 2	Describe the components, architectures, and topologies of Client/Server systems, including two-tier and three-tier models. (Cognitive Knowledge Level: Understand)
CO 3	Explain the services and communication mechanisms used in Client/Server networks such as RPC, DDE, OLE, and CORBA. (Cognitive Knowledge Level: Understand)
CO 4	Apply the concepts of SQL database servers and transaction processing models in Client/Server environments. (Cognitive Knowledge Level: Apply)
CO 5	Illustrate the use of web technologies and web services such as APIs, Microservices, and Web Servers in Client/Server-based applications. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	1					1			2	1	1
CO 2	1	2	2	1	1	2			2	2	1
CO 3	2	2	1		2	2		1	3	1	3
CO 4	2	3	2	3	3	3	2	2	2	3	3
CO 5	2	3	3	3	3	3	2	3	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember			
Understand	70	70	70
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study (Quiz/seminar/microproject/video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction:

Introduction to Client/Server computing - Driving forces behind Client/ Server, Client/ Server development tools, Development of client/server systems, Client/Server security, Organizational Expectations, Improving performance of client/server applications, Single system image, Downsizing and Rightsizing, Advantages of client server computing, Applications of Client/Server.

MODULE 2 (8 hours)

Client/Server Application Components:

Classification of Client/Server Systems- Two-Tier Computing, Middleware, Three-Tier Computing- Model View Controller (MVC), Principles behind Client/Server Systems. Client/Server Topologies. Existing Client/Server Architecture. Architecture for Business Information System.

MODULE 3 (7 hours)

Client/Server Network:

Client- Services, Request for services, RPC, Windows services, Print services, Remote boot services, other remote services, Utility Services. Dynamic Data Exchange (DDE). Object Linking and Embedding (OLE). Common Object Request Broker Architecture (CORBA). Server- Detailed server functionality, Network operating system, Available platforms, Server operating system..

MODULE 4 (7 hours)

Client-Server Database and Transaction Processing::

SQL Database Servers – Server Architecture – Multithread Architecture – Hybrid Architecture – Stored Procedures – Triggers – Client-Server Transaction Processing – Rules of Client-Server Transaction Processing – Transaction Models – Chained and Nested Transactions – Transaction Management Standards – Data Warehousing – Warehousing Techniques – Data Mining.

MODULE 5 (7 hours)

Client/Server Technology and Web Services:

Web Services History. Web Server Technology- Web Server, Web Server Communication, Role of Java for Client/Server on Web. Web Services- MicroServices, APIs, API Gateway, Authentication of users/clients, Tokens/Keys for Authentication, Service Mesh, Message Queues, SaaS, Web Sockets. Client/Server/Browser – Server Technology, Client/Server Technology and Web Applications, Balanced Computing and the Server's Changing Role. Thin client computing - Computing models-Comparison-Computing Environment. Future of client/ server Computing Enabling Technologies, Transformational system.

Text Books

1. Patrick Smith & Steave Guengerich, “Client / Server Computing”, PHI
2. Dawna Travis Dewire, “Client/Server Computing”, TMH
3. Robert Orfali, Dan Harkey and Jerri Edwards: Essential Client/Server Survival Guide, John Wiley & Sons Inc 1996

Reference Books

1. Jeffrey D.Schank, “Novell’s Guide to Client-Server Application & Architecture” Novell Press
2. Robert Orfali, Dan Harkey, Jeri Edwards, Client/Server Survival Guide, Wiley- India Edition, Third Edition
3. W. H. Inman, Developing Client Server Applications, BPB

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction:		7 hours
1.1	Driving forces behind Client/ Server	1
1.2	Client Server development tools	1
1.3	Development of client/server systems	1
1.4	Client/Server security, Organizational Expectations	1
1.5	Improving performance of client/server applications	1
1.6	Single system image, Downsizing and Rightsizing	1
1.7	Advantages and Applications of client server computing	1
Module 2: Client/Server Application Components:		8 hours
2.1	Classification of Client/Server Systems	1
2.2	Open System Standards	1
2.3	Two-Tier Computing	1
2.4	Three-Tier Computing, Middleware	1
2.5	Principles behind Client/Server Systems	1
2.6	Client/Server Topologies	1
2.7	Existing Client/Server Architecture	1

2.8	Architecture for Business Information System	1
Module 3: Client/Server Network:		7 hours
3.1	The client: Services, Request for services, RPC, Windows services, Print services	1
3.2	Remote boot services, Utility Services & Other Services	1
3.3	Dynamic Data Exchange (DDE), Object Linking and Embedding (OLE)	1
3.4	Common Object Request Broker Architecture (CORBA)	1
3.5	The server: Detailed server functionality, the network operating system	2
3.6	Available platforms, the server operating system	1
Module 4: Client-Server Database and Transaction Processing		7 hours
4.1	SQL Database Servers and Server Architecture	1
4.2	Multithread Architecture, Hybrid Architecture	1
4.3	Stored Procedures, Triggers	1
4.4	Client-Server Transaction Processing and Rules of Client-Server Transaction Processing	1
4.5	Transaction Models	1
4.6	Chained Transactions and Nested Transactions, Transaction Management Standards	1
4.7	Data Warehousing, Warehousing Techniques, and Data Mining	1
Module 5: Client/Server Technology and Web Services:		7 hours
5.1	Web Services History , Web Server Technology , Web Server	1
5.2	Web Server Communication , Role of Java for Client/Server on Web	1
5.3	Web Services , MicroServices, APIs, API Gateway, Authentication of users/clients	1
5.4	Tokens/Keys for Authentication ,Service Mesh, Message Queues	1
5.5	SaaS, Web Sockets ,Client/Server Technology and Web Applications	1
5.6	Balanced Computing and the Server's Changing Role ,Thin client computing , Computing models, Computing Environment	1
5.7	Future of client/ server Computing Enabling Technologies, Transformational system	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the fundamental concepts and characteristics of Client/Server computing.
2. Discuss the advantages and applications of Client/Server computing in modern computing environments.

Course Outcome 2 (CO 2):

1. Describe the architecture of two-tier and three-tier client/server computing models.
2. Explain the components and topologies used in Client/Server systems.

Course Outcome 3 (CO 3):

1. Explain the communication mechanisms used in Client/Server networks such as RPC, DDE, OLE, and CORBA.
2. Describe the different services provided by clients and servers in a Client/Server network.

Course Outcome 4 (CO 4):

1. Illustrate how SQL database servers support transaction processing in a Client/Server environment.
2. Apply different transaction models such as chained and nested transactions in a Client/Server database system.

Course Outcome 5 (CO 5):

1. Illustrate how APIs and Microservices are used in developing Client/Server-based web applications.
2. Apply web server technology and web services in the development of Client/Server web applications.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CS4P44

Course Name: CLIENT SERVER COMPUTING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define Client/Server computing. List its key advantages.
2. Explain the driving forces behind the development of Client/Server systems.
3. Differentiate between two-tier and three-tier Client/Server architecture.
4. What is middleware? Explain its role in Client/Server systems.
5. What is CORBA? State its purpose in distributed systems.
6. What is Remote Procedure Call (RPC)? Explain its role in Client/Server communication.
7. Define data warehousing and data mining.
8. What are stored procedures and triggers in SQL database servers?
9. What is an API Gateway? Mention its functions.
10. Explain the concept of thin client computing.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the fundamental concepts and driving forces behind Client/Server computing. (8)
(b) Discuss the advantages and major applications of Client/Server computing. (6)

OR

12. (a) Explain the concept of a Single System Image in Client/Server computing. (7)
(b) Discuss the strategies used to improve the performance of Client/Server applications. (7)
13. (a) Explain the classification of Client/Server systems with emphasis on two-tier computing architecture. (7)
(b) Describe the three-tier architecture and explain how it improves scalability and performance. (7)

OR

14. (a) Explain the principles behind Client/Server systems. (8)
(b) Describe different Client/Server topologies used in distributed environments. (6)
15. (a) Explain the different services provided by clients in a Client/Server network. (7)
(b) Describe the process of request for services and the role of Remote Procedure Call (RPC) in Client/Server communication. (7)

OR

16. (a) Explain the detailed functionality of a server in a Client/Server network. (8)
(b) Discuss the role of a Network Operating System (NOS) in supporting Client/Server environments. (6)
17. (a) What are stored procedures and triggers? Explain their role in SQL database servers. (7)
(b) Discuss the rules of Client-Server transaction processing. (7)

OR

18. (a) Explain different transaction models used in Client-Server systems.. (8)
(b) Describe chained transactions and nested transactions with suitable (6) examples.
19. (a) Explain the concepts of Microservices and APIs in web services. (6)
(b) Discuss the role of an API Gateway and the methods used for authentication using tokens/keys. (8)

OR

20. (a) Explain Client/Server/Browser technology and its role in web applications. (7)
(b) Discuss the concept of balanced computing and the changing role of servers. (7)

B24CS4G21	INTRODUCTION TO DATABASES	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course introduces the fundamental concepts of database systems, including data organization, relational models, and SQL for data manipulation. It covers essential topics such as database design, normalization, and transaction management to ensure data integrity and consistency. The course also provides an overview of modern NoSQL databases and their applications in handling large-scale and diverse data.

Prerequisites

Basic understanding of Programming in C and Data Structures is recommended

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the fundamental concepts of database systems and basic SQL operations.(Cognitive Knowledge Level: Understand)
CO 2	Apply SQL commands to create, manipulate, and query relational databases using appropriate constraints and joins.(Cognitive Knowledge Level: Apply)
CO 3	Describe relational database design concepts, including ER modeling, functional dependencies, and normalization up to 3NF. (Cognitive Knowledge Level: Understand)
CO 4	Apply concepts of transaction management and basic recovery techniques, to maintain consistency and reliability in database systems.(Cognitive Knowledge Level: Apply)
CO 5	Explain the concepts, types, and applications of NoSQL databases along with their characteristics and limitations.(Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	1						1		1		2
CO 2	3	3	3	3	2	1		2	2	1	
CO 3	1	3	3	3	3	1		1	3	2	1
CO 4	3	3	3	3	3	3	2	3	3	3	2
CO 5	1	3	3	3	3	3	3	3	3	2	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study (Quiz/seminar/microproject/video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction to Databases & SQL Basics:

Data and database concepts, DBMS and its need , File system vs DBMS , Relational model – Tables, rows, columns , Primary key , Foreign key , Basic constraints; Introduction to SQL – CREATE TABLE , INSERT , UPDATE , DELETE , SELECT , WHERE , ORDER BY; Simple JOINS ; Introduction to DBMS tools like MySQL and Oracle Database

MODULE 2 (7 hours)

Relational Design & Intermediate SQL

ER diagrams – Entities and relationships , Converting ER diagram to tables; Functional dependency (basic idea); Normalization up to 3NF (simple examples); SQL subqueries – GROUP BY , HAVING , UNION , Views , Practice problems

MODULE 3 (7 hours)

SQL and Basic Transaction Concepts

Transactions, ACID properties, COMMIT and ROLLBACK; Basic concurrency problems (lost update, dirty read) ; Introduction to indexing (concept only); Simple query performance ideas , GRANT and REVOKE ; More SQL practice

MODULE 4 (7 hours)

Introduction to Transaction Management and Recovery

Serial schedules, Concurrent and Serializable Schedules, Conflict equivalence and conflict serializability, Recoverable and cascade-less schedules, Locking, Two-phase locking and its variations. Log-based recovery, Deferred database modification, check-pointing (basic concepts only).

MODULE 5 (8 hours)

Introduction to NoSQL Databases

Introduction to NoSQL Concepts, Types of NoSQL Databases, CAP Theorem, BASE Properties, Use Cases and Limitations of NoSQL Databases, Multidimensional Data Models, Main Characteristics of Key-Value Databases (examples from: Redis), Document Databases (examples from: MongoDB), Column-Family Databases (examples from: Apache Cassandra), and Graph Databases (examples from: ArangoDB)

Text Books

1. Fundamentals of Database Systems, Elmasri, Navathe, Pearson 7/e.
2. Making the Sense of NoSQL: A guide for Managers and rest of us, Dan McCreary and Ann Kelly, Manning, 2014.

Reference Books

1. Steven Feuerstein — Oracle PL/SQL Programming, 6/e, O'Reilly
2. Alexis Leon and Mathews Leon, Database Management Systems
3. Pramod Sadalage and Martin Fowler, NoSQL Distilled

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to Databases & SQL Basics		7 hours
1.1	Data and database concepts, DBMS and its need	1
1.2	File system vs DBMS, Relational model – Tables, rows, columns	1
1.3	Relational model – Primary key, Foreign key, Basic constraints	1
1.4	Introduction to SQL – CREATE TABLE, INSERT, UPDATE, DELETE	1
1.5	Introduction to SQL – SELECT, WHERE, ORDER BY	1
1.6	Simple JOINS	1
1.7	Introduction to DBMS tools like MySQL and Oracle Database	1
Module 2: Relational Design & Intermediate SQL		7 hours
2.1	ER diagrams – Entities and relationships	1
2.2	Converting ER diagram to tables	1
2.3	Functional dependency (basic idea)	1
2.4	Normalization up to 3NF (simple examples)	1
2.5	Normalization up to 3NF (simple examples), SQL subqueries – GROUP BY	1

2.6	SQL subqueries – HAVING , UNION , Views ,	1
2.7	Practice problems	1
Module 3: SQL and Basic Transaction Concepts		7 hours
3.1	Transactions and ACID properties	1
3.2	COMMIT and ROLLBACK	1
3.3	Basic concurrency problems(lost update, dirty read)	2
3.4	Introduction to indexing(concept only)	1
3.5	Simple query performance ideas , GRANT and REVOKE	1
3.6	SQL practice	1
Module 4: Introduction to Transaction Management and Recovery		7 hours
4.1	Serial schedules, Concurrent and Serializable Schedule	1
4.2	Conflict equivalence and conflict serializability	1
4.3	Recoverable and cascade-less schedule	1
4.4	Locking, Two-phase locking and its variations.	1
4.5	Log-based recovery	2
4.6	Deferred database modification, check-pointing (basic concepts only).	1
Module 5:Introduction to NoSQL Databases		8 hours
5.1	Introduction to NoSQL Concepts	1
5.2	Types of NoSQL Databases	1
5.3	CAP Theorem, BASE Properties	1
5.4	Use Cases and Limitations of NoSQL Databases, Multidimensional Data Models	1
5.5	Main Characteristics of Key–Value Databases (examples from: Redis)	1
5.6	Main Characteristics of Document Databases (examples from: MongoDB)	1
5.7	Main Characteristics of Column–Family Databases (examples from: Apache Cassandra)	1
5.8	Main Characteristics of Graph Databases (examples from: ArangoDB)	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the need for a database management system. Compare files and databases with suitable examples.
2. Using an example, demonstrate the use of basic SQL commands such as CREATE, INSERT, UPDATE, DELETE, and SELECT.

Course Outcome 2 (CO2):

1. Draw an ER diagram for a given scenario and convert it into a relational schema.
2. Explain functional dependencies and normalization up to 3NF with examples.

Course Outcome 3 (CO3):

1. Define transactions and ACID properties. Explain how isolation levels affect concurrent execution.
2. Demonstrate the use of GRANT and REVOKE in database security with an example.

Course Outcome 4 (CO4):

1. Write PL/SQL programs to demonstrate conditional statements, loops, and cursor operations.
2. Explain the structure of a stored procedure and stored function with examples.

Course Outcome 5 (CO5):

1. Develop a PL/SQL trigger to enforce a business rule using BEFORE or AFTER events.
2. Explain the concept of packages and dynamic SQL. Provide an example of a scenario where dynamic SQL is required.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CS4G21

Course Name: INTRODUCTION TO DATABASES

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define DBMS. Give any three advantages over file systems.
2. Differentiate between file system and DBMS.
3. Explain primary key and foreign key with examples.
4. What is a relational model? Define table, row, and column.
5. Write an SQL query to display all employees ordered by salary in descending order.
6. What is a JOIN? Mention any two types.
7. What is a functional dependency? Give an example.
8. Define ACID properties.
9. What is indexing? State its purpose.
10. What is CAP theorem in NoSQL?

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain DBMS and its need. Compare file system and DBMS. (7)

- (b) Explain relational model concepts: table, row, column, primary key, and foreign key. (7)

OR

12. (a) Explain basic constraints in SQL with examples. (7)
(b) Write SQL commands for CREATE TABLE, INSERT, UPDATE, DELETE, SELECT with examples. (7)
13. (a) Explain ER diagrams with entities and relationships. Convert a simple ER diagram into tables. (7)
(b) Explain normalization up to 3NF with suitable examples. (7)

OR

14. (a) Explain functional dependency with examples. (7)
(b) Write SQL queries using GROUP BY, HAVING, and subqueries. (7)
15. (a) Explain transactions and ACID properties. (7)
(b) Explain concurrency problems: lost update and dirty read. (7)

OR

16. (a) Explain indexing (concept only) and its advantages. (7)
(b) Explain GRANT and REVOKE with examples. (7)
17. (a) Explain serial and concurrent schedules. (7)
(b) Explain conflict serializability and conflict equivalence. (7)

OR

18. (a) Explain two-phase locking (2PL) and its variations. (7)
(b) Explain log-based recovery and checkpointing. (7)
19. (a) Explain NoSQL concepts and types of NoSQL databases. (7)
(b) Explain CAP theorem and BASE properties. (7)

OR

20. (a) Explain key-value, document, column-family, and graph databases with examples. (7)
(b) Discuss use cases and limitations of NoSQL databases. (7)

B24CS4G22	BASICS OF LAN TECHNOLOGIES	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

Study of this course provides the learners with a clear understanding of Local Area Networks (LANs) and data communication principles essential for modern computer networks. This course covers fundamental concepts of network architectures, explores various LAN standards and protocols, data communication techniques, LAN access mechanisms, Ethernet technologies, token-passing networks, ATM LANs and wireless LANs. The course helps learners to understand, design and evaluate networking systems in real-world applications.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain LAN fundamentals, architectures (OSI, TCP/IP), and IEEE 802 protocols. models (Cognitive Knowledge Level: Understand)
CO 2	Analyze data communication concepts, encoding, error control, and multiplexing techniques. (Cognitive Knowledge Level: Analyze)
CO 3	Evaluate LAN access methods like ALOHA, CSMA/CD, and controlled access schemes. (Cognitive Knowledge Level: Apply)
CO 4	Explain Ethernet technologies and token-passing LAN mechanisms. (Cognitive Knowledge Level: Understand)
CO 5	Analyze ATM and wireless LAN technologies including IEEE 802.11 and Bluetooth (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1			1	2	1	2	1		2
CO 2	3	3	1	3	3	3	2	2	3	3	2
CO 3	3	3	3	3	3	3	2	2	3	3	2
CO 4	2	3	3	2	2	3	2	3	3	2	3
CO 5	3	3	3	3	3	3	3	3	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	40	30	30
Understand	50	50	50
Apply	10	20	20
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Overview of LANs and Network Architecture:

Basic LAN Terminology, LAN Standards, How LANs Operate, LAN Topologies, Example Applications of LANs.

Network Architecture Concepts: Basic Concepts of Layering, Layers of the OSI Model, The TCP/IP Architecture, LAN Protocols - IEEE Project 802, The Logical Link Control (LLC) Layer, The Media Access Control (MAC) Sublayer.

MODULE 2 (7 hours)

Data Communication Concepts:

Basic Concepts - Terminology, Signal Characteristics, Signal Impairments, Channel Capacity. Signal Encoding Techniques, Error Detection- Bit Error Rate (BER), Types of Errors, Cyclic Redundancy Check. Error Correction, Data Link Control- Flow Control, Error Control. Transmission Media, Multiplexing - FDM, TDM, WDM.

MODULE 3 (8 hours)

LAN Access Techniques:

Performance Measures and Notation, Random Access Overview, ALOHA - Pure ALOHA, Slotted ALOHA. Carrier Sense Multiple Access (CSMA) - Nonpersistent CSMA, Slotted Nonpersistent CSMA, 1-Persistent CSMA. CSMA with Collision Detection (CSMA/CD), Controlled Access Schemes - Polling, Token Passing. Switched Access Methods, Stability of Slotted ALOHA, Throughput Analysis for CSMA, Nonpersistent CSMA, Slotted Nonpersistent CSMA.

MODULE 4 (7 hours)

Ethernet and Token-Passing LANs:

Ethernet : Nomenclature, Basic Ethernet Concepts, Logical Link Control (LLC), Fast Ethernet - Architecture, Physical Media for 100BASE-T, Network Spans Using 100BASE-T Fast Ethernet. Switched Ethernet, Gigabit Ethernet - Architecture, General Functions, Transmission Media, 10-Gigabit Ethernet

Token-Passing LANs: Classic Token Rings- Token-Ring Operation, IEEE 802.5 Frame Format, Ring Management, Ring Maintenance, Token-Ring Performance Analysis.

MODULE 5 (8 hours)

ATM LANs and Wireless LANs:

ATM LANs - ATM Fundamentals, Use of ATM Cells, ATM Architecture, ATM Layers, ATM Reference, ATM Service Categories- Service Classes, Quality-of-Service Parameters. ATM Adaptation Layer (AAL)- AAL1, AAL2, AAL3/4, AAL5.

Wireless LANs: Overview, Physical Layer Architecture, Wireless LAN Configurations, MAC Layer - IEEE 802.11 MAC Services, MAC Layer Operation, MAC Frame Formats, Spread Spectrum

Wireless LANs- Hopping Spread Spectrum, Frequency-Hopping Spread Spectrum, Direct Sequence Spread Spectrum. Infrared Wireless LANs, Other Physical Layer Protocols - IEEE 802.11b, IEEE 802.11a. Wireless PANs- Bluetooth Technology, Bluetooth Packets

Text Books

1. Gerd Keiser, Local Area Networks, 2/e, Tata McGraw Hill
2. S. K. Basandra, S. Jaiswal, Local Area Networks, 5/e, Galgotia Publications Pvt Ltd.

Reference Books

1. Andrew S. Tanenbaum, Computer Networks, 4/e, PHI (Prentice Hall India).
2. Fred Halsall, Computer Networking and the Internet, 5/e.
3. James F. Kurose, Keith W. Ross, Computer Networking: A Top-Down Approach, 6/e.
4. Keshav, An Engineering Approach to Computer Networks, Addison Wesley, 1998.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Overview of LANs and Network Architecture		6 hours
1.1	Basic LAN Terminology, LAN Standards	1
1.2	How LANs Operate, LAN Topologies, Example Applications of LANs.	1
1.3	Network Architecture Concepts: Basic Concepts of Layering, Layers of the OSI Model	1
1.4	The TCP/IP Architecture,	1
1.5	AN Protocols - IEEE Project 802, The Logical Link Control (LLC) Layer	1
1.6	The Media Access Control (MAC) Sublayer.	1
Module 2: Data Communication Concepts		7 hours
2.1	Basic Concepts - Terminology, Signal Characteristics	1
2.2	Channel Capacity, Signal Encoding Techniques	1

2.3	, Error Detection- Bit Error Rate (BER)	1
2.4	Types of Errors, Cyclic Redundancy Check.	1
2.5	. Error Correction, Data Link Control- Flow Control, Error Control.	1
2.6	Transmission Media	1
2.7	Multiplexing - FDM, TDM, WDM.	1
Module 3: LAN Access Techniques		8 hours
3.1	Performance Measures and Notation, Random Access Overview	1
3.2	ALOHA - Pure ALOHA, Slotted ALOHA	1
3.3	CSMA - Nonpersistent CSMA, Slotted Nonpersistent CSMA	1
3.4	1-Persistent CSMA, CSMA with Collision Detection (CSMA/CD)	1
3.5	Controlled Access Schemes - Polling, Token Passing	1
3.6	Switched Access Methods	1
3.7	Stability of Slotted ALOHA	1
3.8	Throughput Analysis for CSMA, Nonpersistent CSMA, Slotted Nonpersistent CSMA	1
Module 4: Ethernet and Token-Passing LANs		7 hours
4.1	Ethernet Nomenclature, Basic Ethernet Concepts, Logical Link Control (LLC)	1
4.2	Fast Ethernet - Architecture, Physical Media for 100BASE-T, Network Spans	1
4.3	Switched Ethernet, Gigabit Ethernet - Architecture, General Functions	2
4.4	Gigabit Ethernet Transmission Media, 10-Gigabit Ethernet	1
4.5	Token Ring - Operation, IEEE 802.5 Frame Format	1
4.6	Ring Management, Ring Maintenance, Token Ring Performance Analysis	1
Module 5: ATM LANs and Wireless LANs		8 hours
5.1	ATM Fundamentals, Use of ATM Cells, ATM Architecture	1
5.2	ATM Layers, ATM Reference Model	1
5.3	ATM Service Categories, Service Classes, Quality-of-Service Parameters	1
5.4	ATM Adaptation Layer (AAL) - AAL1, AAL2, AAL3/4, AAL5	1
5.5	Wireless LAN Overview, Physical Layer Architecture, WLAN Configurations	1

5.6	IEEE 802.11 MAC Layer - MAC Services, MAC Operation, Frame Formats	1
5.7	Spread Spectrum Techniques - FHSS, DSSS, Infrared Wireless LANs	1
5.8	IEEE 802.11a, 802.11b, Wireless PANs - Bluetooth Technology, Bluetooth Packets	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the OSI model in detail, describing the functions of each layer with a neat diagram.
2. Describe the TCP/IP architecture and explain how it differs from the OSI model. Also discuss the role of IEEE 802 standards in LANs.

Course Outcome 2 (CO 2):

1. Analyze different error detection techniques, including CRC, and compare their effectiveness.
2. Compare and analyze multiplexing techniques (FDM, TDM, WDM) based on bandwidth utilization and applications.

Course Outcome 3 (CO 3):

1. A network with heavy traffic experiences frequent collisions. Which access method (ALOHA, CSMA/CD, or Token Passing) would you recommend? Justify your choice.
2. Given a network scenario with multiple nodes trying to transmit data simultaneously, illustrate how CSMA/CD handles collisions and improves performance step-by-step.

Course Outcome 4 (CO 4):

1. Explain Ethernet technologies, including Fast Ethernet and Gigabit Ethernet, with architecture and features.
2. Describe Token Ring networks, including token passing mechanism and frame format.

Course Outcome 5 (CO 5):

1. A company requires a network that supports real-time voice and video with guaranteed QoS. Explain how ATM can be used to meet this requirement.
2. Design a wireless LAN for a college campus. Explain which IEEE 802.11 standards and spread spectrum techniques you would choose and why.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CS4G22

Course Name: BASICS OF LAN TECHNOLOGIES

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define LAN topology. List any two types.
2. Explain the role of the MAC sublayer.
3. Define Bit Error Rate (BER).
4. What is flow control in data communication?
5. What is the difference between Pure ALOHA and Slotted ALOHA?
6. What are performance measures in LAN? Give any two.
7. Define Ethernet.
8. What is token passing in LANs?
9. Define Asynchronous Transfer Mode (ATM).
10. Name the different types of AAL (AAL1–AAL5).

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the OSI model in detail, describing the function of each layer. (8)

- (b) Explain the IEEE 802 LAN standards and their significance in networking. (6)

OR

12. (a) Describe the Logical Link Control (LLC) layer and MAC sublayer in detail. Highlight their differences. (7)
(b) Describe the TCP/IP architecture in detail, including its layers and functions. (7)
13. (a) Explain the basic concepts and terminology of data communication, including components and modes of communication. (7)
(b) Explain data link control mechanisms, including flow control and error control techniques. (7)

OR

14. (a) Explain error detection techniques, focusing on Cyclic Redundancy Check (CRC) with an example. (7)
(b) Explain multiplexing techniques: Frequency Division Multiplexing (FDM), Time Division Multiplexing (TDM), and Wavelength Division Multiplexing (WDM). (7)
15. (a) Describe the working of CSMA/CD and explain how collision detection improves performance. (8)
(b) Discuss switched access methods and their role in modern LANs. (6)

OR

16. (a) Compare random access methods and controlled access methods in terms of efficiency, delay, and complexity. (7)
(b) Explain performance measures and notation used in LAN access techniques. (7)
17. (a) Explain Ethernet nomenclature and basic Ethernet concepts in detail. (5)
(b) Describe the architecture of Fast Ethernet and explain the physical media used in 100BASE-T. (9)

OR

18. (a) Discuss the architecture and general functions of Gigabit Ethernet, including transmission media. (8)
(b) Explain Token Ring networks, including token-ring operation and its working principle. (6)
19. (a) Explain ATM service categories, service classes, and QoS parameters in detail. (7)
(b) Provide an overview of Wireless LANs, including architecture and configurations. (7)

OR

20. (a) Explain the IEEE 802.11 MAC layer, including MAC services and operation. (7)
(b) Explain Spread Spectrum techniques: Frequency-Hopping Spread Spectrum (FHSS) and Direct Sequence Spread Spectrum (DSSS). (7)

B24CS4G23	INTRODUCTION TO MOBILE COMPUTING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4		2024

Preamble

The purpose of this course is to prepare learners to understand the functionalities and design considerations of mobile computing. The course content is designed to cover the mobile computing architecture, features of different communication systems and major elements of mobile security and next generation computer systems. This course enables the learners to acquire advanced concepts on mobile and ad-hoc networks.

Prerequisites

A good knowledge of data communication and computer networks.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the concepts of mobile computing, including its functions, devices, middleware, gateways, applications, services, limitations, and the architecture. (Cognitive Knowledge Level: Understand)
CO 2	Explain the evolution of telephony, multiple access procedures, satellite communication systems, frequency allocation, authentication, and security. (Cognitive Knowledge Level: Understand)
CO 3	Explain the architecture, services, and applications of SMS and GPRS in mobile communication systems. (Cognitive Knowledge Level: Understand)
CO 4	Analyse the concepts, architecture, mobility, security, and applications of WLAN, along with WLL architecture, HIPERLAN, and comparison of WiFi and 3G technologies. (Cognitive Knowledge Level: Analyse)
CO 5	Analyze the security issues and techniques in mobile computing and evaluate the concepts and technologies used in next generation networks and multimedia broadcast services. (Cognitive Knowledge Level: Analyze)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1		1		2			1		
CO 2	3	2	1								
CO 3	1	2	2	1	3			3	1	3	
CO 4		2	2	3		3	3		3		3
CO 5	1	1	2	2	3			3	1	3	2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self Study(Quiz/ Seminar/ Microproject/ Video Assignments/ Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Mobile Computing Architecture

Introduction to mobile computing – Functions, Devices, Middleware and gateways, Applications and services, Limitations. Mobile computing architecture – Internet: The ubiquitous network, Three-tier architecture, Design considerations for mobile computing.

MODULE 2 (8 hours)

Communication Systems

Mobile computing through telephony - Evolution of telephony, Multiple access procedures - Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA), Code Division Multiple Access (CDMA), Space Division Multiple Access (SDMA). Satellite communication systems – Basics, Applications, Geostationary Earth Orbit (GEO), Low Earth Orbit (LEO), Medium Earth Orbit (MEO), Satellite phones. Mobile computing through telephone – Interactive Voice Response (IVR) architecture, Overview of voice software, Developing an IVR application. Global System for Mobile Communication (GSM) - Introduction, Architecture, Entities, Call routing, Mobility management, Frequency allocation, Authentication and security.

MODULE 3 (8 hours)

Short Messaging Service and General Packet Radio Service

Short Message Service (SMS) – Strengths, Architecture, Value added services, Accessing the SMS bearer. General Packet Radio Service (GPRS) – Architecture, Network operations, Data services, Applications, Limitations, Billing and charging.

MODULE 4 (7 hours)

Wireless Local Area Networks

Wireless Local Area Network (WLAN) - Advantages, Evolution, Applications, Architecture, Mobility, Security, Deploying WLAN. Wireless Local Loop (WLL) – Architecture. High Performance Radio Local Area Network (HIPERLAN). WiFi Vs 3G.

MODULE 5 (7 hours)

Mobile Security and Next Generation Networks

Security issues in mobile computing - Information security, Security techniques and algorithms, Security protocols. Next generation networks – The Converged Scenario, Narrowband to broadband, Orthogonal Frequency Division Multiplexing (OFDM), Multi Protocol Label Switching (MPLS), Wireless Asynchronous Transfer Mode (WATM), Multimedia broadcast services.

Text Books

1. Asoke K. Talukder, Hasan Ahmad, Roopa R Yavagal, Mobile Computing Technology- Application and Service Creation, 2nd Edition, McGraw Hill Education.
2. Schiller J., Mobile Communications, 2/e, Pearson Education, 2009.

Reference Books

1. Andrew S. Tanenbaum, Computer Networks, 6/e, PHI.
2. Theodore S. Rappaport, Wireless Communications Principles and Practice, 2/e, PHI, New Delhi, 2004.
3. Curt M. White, Fundamentals of Networking and Communication 7/e, Cengage learning.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Mobile Computing Architecture		6 hours
1.1	Introduction to mobile computing – Functions, Devices, Middleware and gateways	1
1.2	Applications, services, limitations, Internet: The ubiquitous network	1
1.3	Three-tier architecture (Lecture 1)	1
1.4	Three-tier architecture (Lecture 2)	1
1.5	Design considerations for mobile computing (Lecture 1).	1
1.6	Design considerations for mobile computing (Lecture 2)	1
Module 2: Communication Systems		8 hours
2.1	Evolution of telephony, Multiple access procedures – FDMA, TDMA, CDMA, SDMA	2
2.2	Satellite communication systems – GEO, MEO, LEO, Satellite phones	1
2.3	Interactive Voice Response (IVR) architecture, Overview of voice software, Developing an IVR application (Call flow diagram)	1
2.4	Introduction to GSM,Architecture	1
2.5	GSM entities, Call routing	1
2.6	Mobility management	1

2.7	Frequency allocation, Authentication and security	1
Module 3:Short Messaging Service and General Packet Radio Service		8 hours
3.1	SMS Strengths, Architecture, Short Message Mobile Terminated (SM MT) and Short Message Mobile Originated (SM MO) messages	1
3.2	SMS Architecture - Operator-centric pull, operator-independent push/pull, Value added services	1
3.3	Accessing the SMS bearer (Lecture 1)	1
3.4	Accessing the SMS bearer (Lecture 2)	1
3.5	GPRS architecture	1
3.6	Network operations	1
3.7	Data services, Applications	1
3.8	Limitations, Billing and charging	1
Module 4: – Wireless Local Area Networks		7 hours
4.1	WLAN Advantages, Evolution, Applications	1
4.2	WLAN Architecture (Lecture 1)	1
4.3	WLAN Architecture (Lecture 2)	1
4.4	Mobility, Security	1
4.5	Deploying WLAN	1
4.6	WLL Architecture, HIPERLAN	1
4.7	WiFi Vs 3G	1
Module 5:Mobile Security and Next Generation Networks		7 hours
5.1	Information security – Attacks, Components	1
5.2	Security techniques and algorithms – Stream Vs Block cipher, Symmetric Vs Asymmetric cryptography	1
5.3	Security techniques and algorithms – RSA, Diffie Hellman Key exchange	1
5.4	Security protocols – Secure Socket Layer, Transport Layer Security, Wireless Transport Layer Security	1
5.5	The Converged Scenario, Narrowband to broadband	1
5.6	Orthogonal Frequency Division Multiplexing (OFDM) and Multi Protocol Label Switching (MPLS)	1
5.7	Wireless Asynchronous Transfer Mode (WATM) and Multimedia broadcast services	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Describe the design considerations in mobile computing.
2. Give five examples of mobile computing applications.

Course Outcome 2 (CO 2):

1. Draw a call flow diagram for a theatre ticket booking system.
2. Illustrate the GSM architecture with figure.

Course Outcome 3 (CO 3):

1. Illustrate the billing and charging services in GPRS.
2. Describe the SMS architecture.

Course Outcome 4 (CO 4):

1. Compare IEEE 802.11, HIPERLAN with respect to their ad-hoc capabilities.
2. Discuss the security mechanism used in WLAN.

Course Outcome 5 (CO 5):

1. With the help of a suitable example, show the working of Diffie-Hellman key exchange algorithm.
2. Bob chooses 7 and 11 as two prime numbers and chooses e as 13. Find an appropriate value for d and decrypt the plaintext 5 send by Alice to Bob.
3. Describe the significance of Orthogonal Frequency Division Multiplexing (OFDM) in next generation networks
4. Describe WATM and Multimedia broadcast services.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CS4G23

Course Name: INTRODUCTION TO MOBILE COMPUTING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Explain the different types of middleware and gateways required in mobile computing.
2. List any six limitations of mobile computing.
3. Compare and contrast the satellite systems – GEO, LEO and MEO.
4. How is frequency allocation done in GSM?
5. What are the various strengths of SMS?
6. How is billing and charging done in GPRS?
7. What are the different types of Wireless LANs?
8. Describe the architecture of a Wireless Local Loop.
9. Explain the key features of TLS protocol.
10. How are attacks classified?

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Describe any four mobile computing functions. (4)

- (b) Explain the three-tier architecture of mobile computing with figure. (10)

OR

12. (a) Describe the significance and functions of core, edge and access network. (6)
(b) Explain the terms (i) Client Context Manager (ii) Policy Manager (iii) Security Manager (iv) Adaptability Manager (8)
13. (a) Why is multiple access important? With the help of suitable examples, explain the various multiple access techniques. (7)
(b) Describe the different algorithms used for security and authentication in GSM. (7)

OR

14. (a) Show how call routing is done in GSM. Give an example. (7)
(b) Explain the process of handover. How does handover differ from roaming? (7)
15. (a) With the help of neat sketches, explain the difference between Short Message Mobile Terminated (SM MT) and Short Message Mobile Originated (SM MO) messages. (6)
(b) Explain the network operations in GPRS. (8)

OR

16. (a) How does operator-centric pull differ from operator-independent push and pull? (7)
(b) Describe the data services and applications of GPRS. (7)
17. (a) Compare the HIPERLAN and OSI layered architecture. (4)
(b) Explain the 802.11 architecture. (10)

OR

18. (a) Compare 3G and WiFi. (7)
(b) Explain the HIPERLAN communication models with suitable diagrams. (7)
19. (a) Given $p = 7$, $q = 17$ and $e = 5$. Find the value of d and also encrypt the message $P = 65$ using RSA. (7)
(b) Explain the role of MPLS in service provisioning. (7)

OR

20. (a) With the help of a suitable example, show the working of Diffie-Hellman key exchange algorithm. (7)
(b) Explain the features of any three multimedia broadcast services. (7)

B24HU4T04	DISASTER MANAGEMENT AND INDUSTRIAL SAFETY	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course explains how disasters happen and how to reduce risk through planning, preparedness, response, and recovery. It also covers industrial safety—identifying hazards, using basic assessment tools, and applying safety rules to prevent accidents.

Prerequisite

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Understand the fundamental concepts, classifications, and phases of the disaster management cycle and their impacts on society. (Cognitive Knowledge Level: Understand)
CO 2	Apply remote sensing resolutions and risk analysis concepts to identify and predict hazards at various administrative levels. (Cognitive Knowledge Level: Apply)
CO 3	Utilize hazard identification techniques like HAZOP and FMEA to assess industrial risks and causation theories. (Cognitive Knowledge Level: Apply)
CO 4	Analyze industrial safety standards, fire safety protocols, and statutory provisions like the Factories Act and OSHA. (Cognitive Knowledge Level: Analyze)
CO 5	Evaluate ethical responsibilities and lessons from historical case studies to develop sustainable disaster prevention strategies. (Cognitive Knowledge Level: Evaluate)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	1			3					1
CO 2	3	3	2	2	3	1					1
CO 3	3	3	3	2	2	2	1	2			1
CO 4			1			3	2	2	3	2	1
CO 5	1	2	2	1		3	3		2		2

Assessment Pattern

Bloom's Category	Continuous Assessment (% Marks)		End Semester Examination (% Marks)
	Test 1	Test 2	
Remember	20	10	10
Understand	30	20	20
Apply	30	25	25
Analyse	20	25	20
Evaluate		20	15
Create			10

Examination Details

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question carries 14 marks and can have maximum 2 sub-divisions.

SYLLABUS

MODULE 1 (8 hours)

Introduction to Disaster Management:

Disaster: definition, scope, and importance. Classification of disasters: natural and man-made. Natural disasters: earthquake, flood, cyclone, drought, landslide, tsunami – causes and effects. Man-made disasters: industrial accidents, chemical disasters, nuclear disasters, fire hazards – impacts of disasters on human life, infrastructure, economy, and environment. Disaster management cycle: prevention, mitigation, preparedness, response, recovery.

MODULE 2 (8 hours)

Disaster Preparedness, Mitigation, and Response:

Disaster risk reduction concepts. Hazard, vulnerability, and risk analysis. Disaster preparedness planning at community, district, state, and national levels. Early warning systems and forecasting methods. Emergency response planning, evacuation procedures, and shelter management. Role of government agencies (NDMA, SDMA, DDMA).

MODULE 3 (7 hours)

Industrial Safety and Hazard Identification:

Introduction to industrial safety and accident prevention. Types of industrial hazards: mechanical, electrical, chemical, biological, radiation, ergonomic. Causes of industrial accidents and accident causation theories. Hazard identification techniques: HAZOP, FMEA. Safety management systems and safety culture. Risk assessment and control measures.

MODULE 4(7 hours)

Industrial Safety Practices and Legislation

Safety legislation and statutory provisions – Hearing Conservation Programme: principles, implementation, and monitoring. Fire safety: causes of fire, fire prevention, detection, and fire-fighting systems – Electrical safety and machine guarding – Personal Protective Equipment (PPE): types, selection, and usage.

MODULE 5 (6 hours)

Environmental Safety and Case Studies

Environmental pollution due to industrial activities – solid, liquid, and hazardous waste management – sustainable industrial safety practices. Case studies: Bhopal Gas Tragedy, Chernobyl Nuclear Disaster, Fukushima Disaster – lessons learned and preventive strategies.

Text Books

1. Harsh K. Gupta, *Disaster Management*, Universities Press (Orient Black Swan), 2003. ISBN: 9788173714566.
2. John R. Jensen, *Remote Sensing of the Environment: An Earth Resource Perspective*, 2nd Edition, Pearson Prentice Hall, Upper Saddle River, NJ, 2007. ISBN: 9780131889507.
3. K. U. Mistry, *Fundamentals of Industrial Safety and Health (Vol. 1 & 2)*, 4th Edition, Siddharth, 2022.

Reference Books

1. Harsh K. Gupta, *Disaster Management*, Universities Press.
2. Thomas M. Lillesand, Ralph W. Kiefer, Jonathan W. Chipman, *Remote Sensing and Image Interpretation*, Wiley.
3. R. K. Jain, Sunil S. Rao, *Industrial Safety, Health and Environment Management Systems*, Khanna Publishers.
4. Roger L. Brauer, *Safety and Health for Engineers*, Wiley.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
MODULE 1: Introduction to Disaster Management		8 hours
1.1	Disaster: definition, scope, and importance. Classification of disasters: natural and man-made. Natural disasters: earthquake, flood, cyclone, drought, landslide, tsunami – causes and effects.	2
1.2	Man-made disasters: industrial accidents, chemical disasters, nuclear disasters, fire hazards.	2
1.3	Impacts of disasters on human life, infrastructure, economy, and environment.	2
1.4	Disaster management cycle: prevention, mitigation, preparedness, response, recovery.	2
MODULE 2: Disaster Preparedness, Mitigation, and Response		8 hours
2.1	Disaster risk reduction concepts. Hazard, vulnerability, and risk analysis.	2
2.2	Disaster preparedness planning at community, district, state, and national levels.	2
2.3	Early warning systems and forecasting methods.	2
2.4	Emergency response planning, evacuation procedures, and shelter management.	1
2.5	Role of government agencies (NDMA, SDMA, DDMA).	1
MODULE 3: Industrial Safety and Hazard Identification		7 hours
3.1	Introduction to industrial safety and accident prevention. Types of industrial hazards: mechanical, electrical, chemical, biological, radiation, ergonomic.	2

3.2	Causes of industrial accidents and accident causation theories.	1
3.3	Hazard identification techniques: HAZOP, FMEA, Fault Tree Analysis.	2
3.4	Safety management systems and safety culture. Risk assessment and control measures.	2
MODULE 4: Industrial Safety Practices and Legislation		7 hours
4.1	Safety Legislation & Statutory Provisions, Hearing Conservation Programme.	2
4.2	Fire Safety Fundamentals.	1
4.3	Fire-Fighting Systems and Electrical Safety.	2
4.4	Machine Guarding & Personal Protective Equipment (PPE).	2
MODULE 5: Environmental Safety and Case Studies		6 hours
5.1	Environmental pollution due to industrial activities – solid, liquid, and hazardous waste management.	1
5.2	Sustainable industrial safety practices – ethical responsibilities of engineers in disaster prevention and industrial safety.	2
5.3	Case studies: Bhopal Gas Tragedy, Chernobyl Nuclear Disaster, Fukushima Disaster.	2
5.4	Lessons learned and preventive strategies.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. Define “disaster” and classify disasters into natural and man-made categories. Explain with suitable examples relevant to civil infrastructure.
2. Explain the Disaster Management Cycle (DMC) and discuss how each phase (prevention, mitigation, preparedness, response, recovery) reduces impacts on human life, infrastructure, economy, and environment.

Course Outcome 2 (CO 2):

1. A district experiences repeated flooding. Using the concept $\text{Risk} = \text{Hazard} \times \text{Vulnerability}$, explain how you would identify high-risk zones and prioritize mitigation measures at community and district levels.

2. Explain spatial, spectral, temporal, and radiometric resolutions in remote sensing. For each, mention one disaster-management application (flood/cyclone/landslide monitoring).

Course Outcome 3 (CO 3):

1. In a chemical storage facility, outline the steps to perform a HAZOP study and demonstrate how deviations (e.g., “more pressure”, “less flow”) can lead to accident scenarios.
2. Perform an FMEA framework for an industrial system (any one: boiler, conveyor belt, electrical panel, solvent storage tank): identify failure modes, effects, severity/occurrence/detection, and suggest controls.

Course Outcome 4 (CO 4):

1. Discuss key provisions of the Factories Act related to safety, health, and welfare. Explain how statutory compliance improves industrial safety performance.
2. For an industrial workshop/lab building, prepare a fire safety and electrical safety plan covering fire triangle, detection & suppression systems, safe evacuation, LOTO concept, and machine guarding.

Course Outcome 5 (CO 5):

1. Compare the engineering and governance lessons learned from any two case studies: Bhopal Gas Tragedy, Chernobyl, Fukushima. Focus on root causes, systemic failures, and preventable gaps.
2. As a consulting engineer, propose a sustainable disaster prevention strategy for an industrial township (or campus): include ethical responsibilities, waste/pollution control, emergency preparedness, and “build back better” principles.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B. TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24HU4T04

Course Name: DISASTER MANAGEMENT AND INDUSTRIAL SAFETY

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define disaster and explain the scope and importance of disaster management.
2. Classify disasters into natural and man-made and list any three examples of each.
3. Define hazard, vulnerability, and risk and write the relation between them.
4. List the roles of NDMA, SDMA, and DDMA in disaster preparedness and response.
5. List any six industrial hazards and give one example for each (mechanical/electrical/chemical etc.).
6. Differentiate HAZOP and FMEA (any three points).
7. State any three key safety provisions of the Factories Act related to health and welfare.
8. Explain the fire triangle and list any three fire prevention measures in industries.
9. List types of industrial wastes (solid/liquid/hazardous) and mention one safe disposal method for each.
10. Write any three ethical responsibilities of engineers in disaster prevention and industrial safety.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the causes and effects of any two natural disasters. (7)
(b) Explain the Disaster Management Cycle and describe how each phase reduces impacts. (7)

OR

12. (a) Explain man-made disasters with reference to industrial accidents/ chemical/ nuclear/ fire hazards—causes and impacts. (7)
(b) Explain spatial, spectral, temporal, and radiometric resolutions and mention one application of each in disaster studies. (7)
13. (a) Explain Disaster Risk Reduction (DRR) and describe risk analysis using hazard–vulnerability concepts. (7)
(b) Prepare a community-level disaster preparedness plan (key components and steps). (7)

OR

14. (a) Explain early warning systems and forecasting methods for any one hazard. (7)
(b) Describe evacuation procedures and shelter management for a district-level emergency. (7)
15. (a) Explain accident causation theories and discuss common causes of industrial accidents. (7)
(b) Explain safety management system and elements of safety culture with suitable measures. (7)

OR

16. (a) Conduct a HAZOP outline for a chemical storage/processing unit: nodes, deviations, causes, consequences, safeguards. (7)
(b) Prepare an FMEA for an industrial system (any one: electrical panel/boiler/conveyor/solvent tank): failure modes, effects, and controls. (7)
17. (a) Discuss major Factories Act provisions for safety, health, and welfare with examples. (7)
(b) Explain OSHA principles and how they improve workplace safety performance. (7)

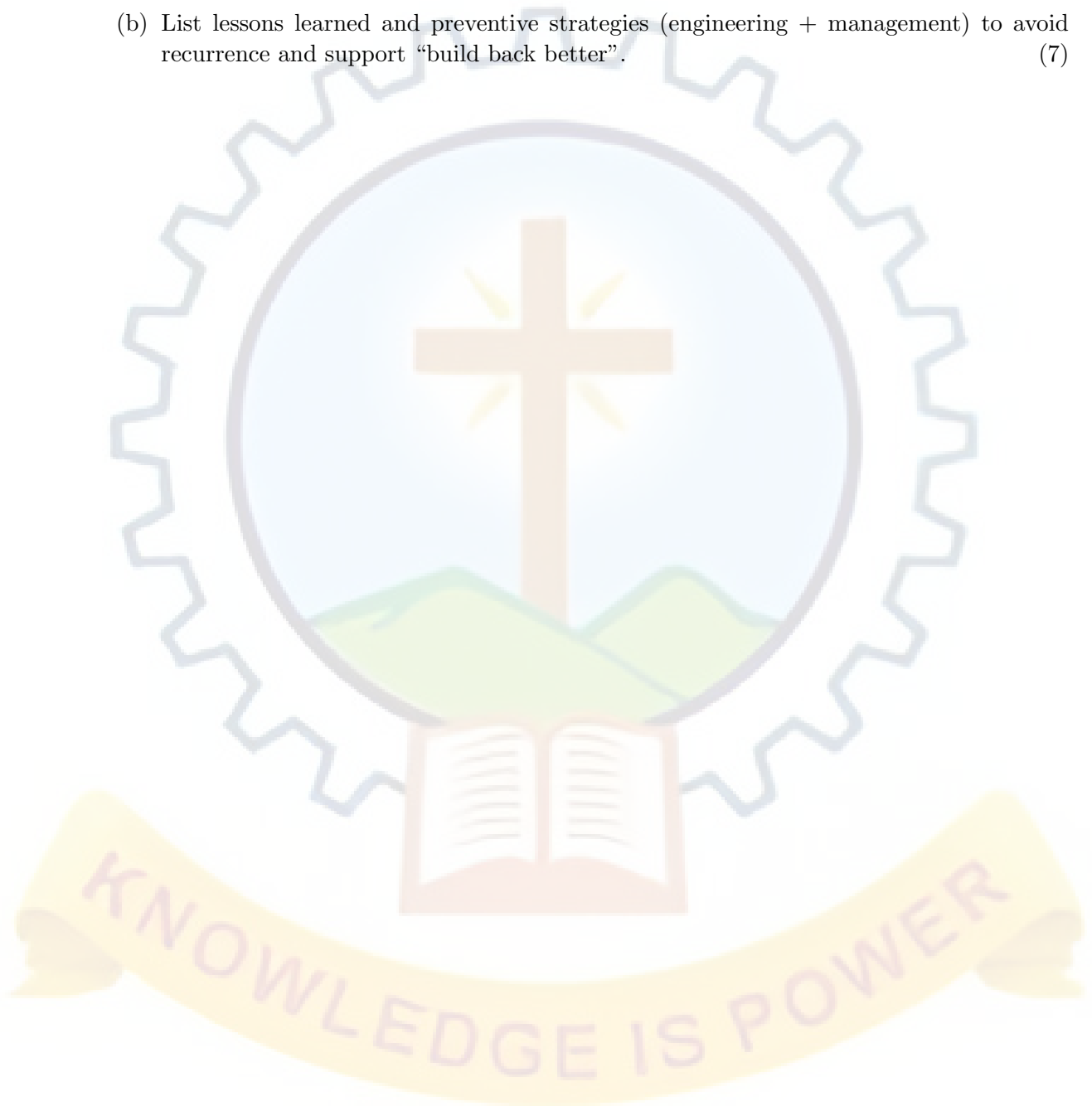
OR

18. (a) Explain fire detection and fire-fighting systems (alarms, extinguishers, hydrants/sprinklers) and a basic emergency plan. (7)
(b) Explain electrical safety and machine guarding, and discuss PPE types, selection, and correct usage. (7)
19. (a) Explain industrial pollution pathways and describe solid, liquid and hazardous waste management practices. (7)

- (b) Explain sustainable industrial safety practices and the ethical role of engineers in prevention. (7)

OR

20. (a) Write a case study on Bhopal / Chernobyl / Fukushima (any one): causes, failures, impacts. (7)
- (b) List lessons learned and preventive strategies (engineering + management) to avoid recurrence and support “build back better”. (7)



B24CS4L11	COMPUTER GRAPHICS LAB	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	3	2		

Preamble

This laboratory provides structured practical experience in implementing fundamental computer graphics algorithms and geometric operations essential for rendering and visual computation . It includes hands-on exercises on drawing algorithms, 2D/3D transformations, clipping techniques, animation routines, and interactive graphics programming . The lab equips students with the technical skills necessary to develop graphical modules and visualization systems for advanced applications in graphics, multimedia, and virtual environments .

Prerequisites

Basic Knowledge in Computer Graphics and Image Processing **Course Outcomes**

After the completion of the course the student will be able to

CO 1	Understand fundamental algorithms for drawing graphical primitives such as lines, circles, and polygons. (Cognitive Knowledge Level: Understand)
CO 2	Apply two-dimensional and three-dimensional geometric transformations for object manipulation. (Cognitive Knowledge Level: Apply)
CO 3	Implement clipping and viewing algorithms to display visible portions of graphical objects. (Cognitive Knowledge Level: Apply)
CO 4	Develop simple animations and interactive graphical applications using basic graphics functions. (Cognitive Knowledge Level: Apply)
CO 5	Apply region-filling and color-processing techniques to enhance graphical scenes. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2		2	1				1		
CO 2	2	3	2	2	2				1		1
CO 3	2	3	2	2	2				1		2
CO 4	3	3	3	3	3	2		2	3	2	3
CO 5	3	3	3	3	3	2	2	2	3	2	3

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Evaluation	20 marks
- Viva (Self Study)	10 marks
- Output	10 marks
Lab Record	5 marks
Viva-voce/Test	15 marks

SYLLABUS

LIST OF EXPERIMENTS

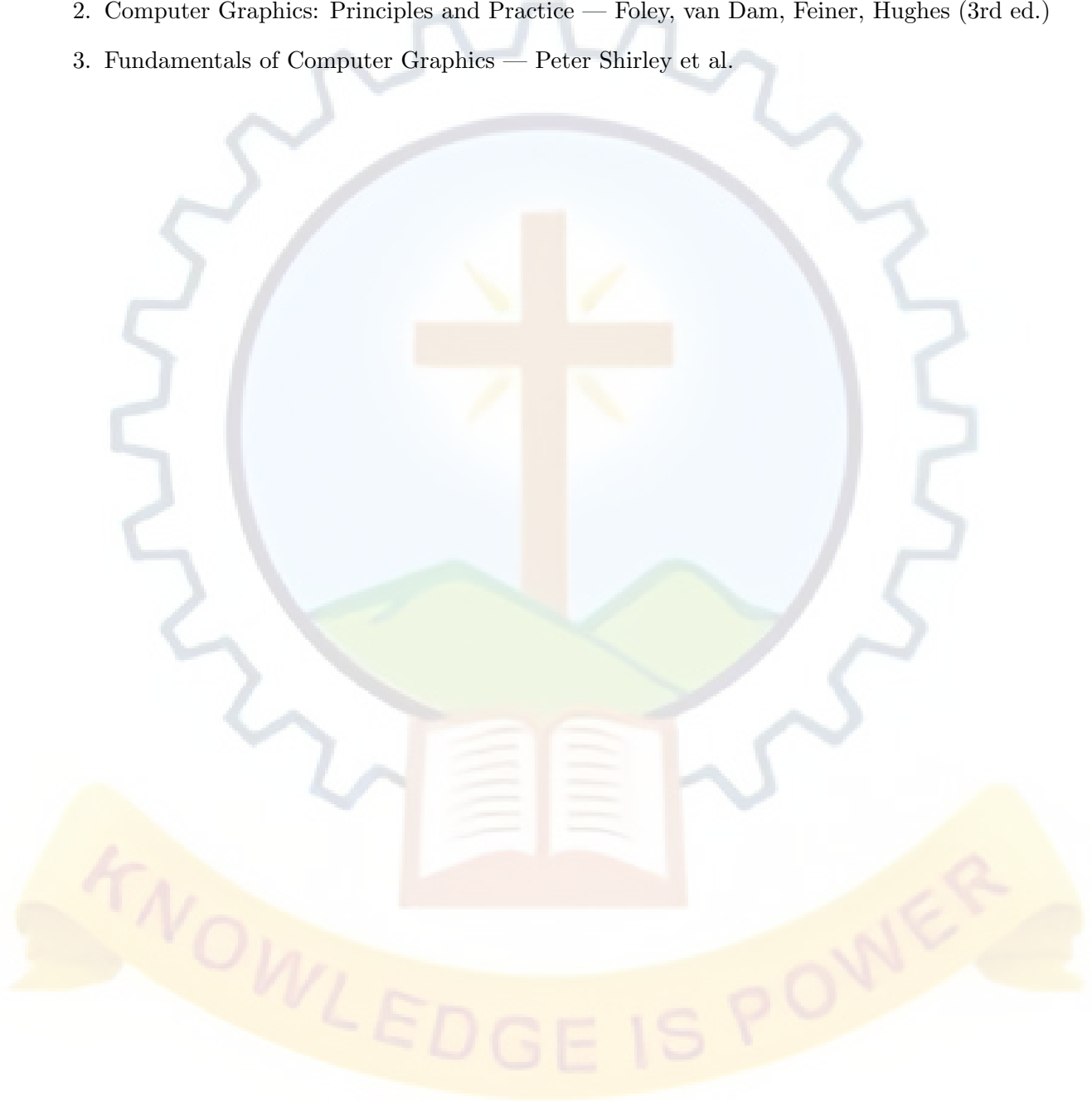
1	A digital art software needs to connect two points on a canvas with a smooth line. Write a program using the DDA algorithm to draw this line segment between the given coordinates.*
2	A railway map application must connect two stations with a clear track line. Write a program to draw the line using Bresenham's method. *
3	A radar system visualization must draw circular ranges around a center point, Write a program to use the Midpoint Circle Algorithm. Implement Bresenham's line and circle drawing algorithms and display plotted pixels stepwise.*
4	A game character must move from one position to another. Write a program using a translation transformation.
5	Write a program using graphics functions to apply a scaling transformation to a logo image. The program should resize the logo proportionally by applying appropriate scaling factors in both X and Y directions. Display the original logo and the scaled logo on the screen.
6	Develop a program to simulate the rotation of a robotic arm joint using 2D rotation transformation equations. The program should take an input angle and apply the rotation matrix to rotate the arm segment about a pivot point. Display both the original and rotated positions of the arm.
7	Write a computer graphics program to display the mirror image of a given object using reflection transformation. The program should apply the appropriate reflection matrix (about the X-axis, Y-axis, or any specified line) and display both the original object and its reflected image.

8	Develop a program to draw a triangle and apply a sequence of geometric transformations:.* a) Translation b) Rotation about a given pivot point c) Scaling
9	Write a program to implement the Cohen–Sutherland Line Clipping algorithm.*
10	A modeling tool clips complex objects during zoom operations so only the visible polygon sections are displayed. Perform this using the Sutherland–Hodgman Polygon Clipping algorithm. *
11	Develop a program to simulate a sunrise that requires a gradual change in background colors and position of the sun using animation sequencing.*
12	Develop a program to show a bouncing ball whose height decreases with every bounce using incremental frame-based animation.*
13	Develop a program to animate a flying airplane moving diagonally across the sky, while its propellers rotate continuously using raster updates.*
14	Develop a program to simulate a traffic signal where the lights smoothly transition from green to yellow to red using frame-sequenced animation.
15	Write a program to visualize a rocket launch by animating the gradual ascent of a rocket along with dynamically changing exhaust flames using raster frame-by-frame animation.
16	Implement an animation of a sunrise and sunset scene by gradually changing the sky color gradient and moving the sun smoothly using raster animation techniques. *
17	Develop a multimedia presentation tool showing a bouncing ball inside a rectangular box, controlled through on-screen buttons labeled Start, Pause, and Reset using raster graphics.
18	Create a simple drawing simulator where the user can move a car left or right across the screen by clicking GUI arrow buttons displayed inside a control panel box.
19	Implement a paint-style application that fills a user-drawn rectangle using the Boundary Fill algorithm, controlled through on-screen Start, Pause, and Reset buttons.*
20	An industrial quality-control system captures grayscale images of metal sheets, where fine cracks appear as thin, low-contrast lines and the images also contain mild sensor noise and uneven illumination. Develop a Python program to preprocess the image using an appropriate smoothing method, then apply Sobel and Prewitt edge detectors to extract crack boundaries, compute and display the horizontal and vertical edge responses separately *
21	A grayscale MRI image contains a suspicious region that appears slightly brighter than surrounding tissues. However, the boundary between normal tissue and the abnormal region is not clearly defined, and intensity varies gradually across the image. Develop a Python program to implement region growing segmentation starting from an appropriate seed point selected inside the suspected tumor region. *

* Mandatory NB: Implement and analyze machine learning models using Python in Visual Studio Code (VS Code) or any other IDE.

Reference Books

1. Computer Graphics with OpenGL — Donald Hearn, M. Pauline Baker & Warren R. Carithers
2. Computer Graphics: Principles and Practice — Foley, van Dam, Feiner, Hughes (3rd ed.)
3. Fundamentals of Computer Graphics — Peter Shirley et al.



B24CS4L12	PROJECT PHASE 1	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	6	2		2024

Preamble

The course 'Project Work' is mainly intended to evoke innovation and invention skills in students. It provides an opportunity to synthesize and apply knowledge and analytical skills to develop a prototype or simulation. Projects should be finalized in thrust areas of the engineering stream or as interdisciplinary efforts, with importance given to addressing societal problems, developing indigenous technologies, and aligning with sustainable development.

Course Objectives

- To apply engineering knowledge in practical problem-solving.
- To foster innovation in the design of products, processes, or systems with consideration for sustainability.
- To develop creative thinking in finding viable, ethical, and socially relevant solutions to engineering problems.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Apply engineering knowledge across domains to model, analyze, and solve real-world problems through practical design, simulation, or prototyping. (Cognitive Knowledge Level: Apply)
CO 2	Develop innovative, sustainable, and socially relevant products, processes, or systems, addressing technology/research gaps while considering ethical, environmental, and SDG-aligned aspects. (Cognitive Knowledge Level: Create)
CO 3	Function effectively as an individual and as a leader/member in diverse teams, demonstrating responsibility, collaboration, and task execution in multidisciplinary settings. (Cognitive Knowledge Level: Apply)
CO 4	Plan, schedule, and execute project tasks efficiently using available resources within timelines while adhering to ethical, professional, and budgetary norms. (Cognitive Knowledge Level: Apply)
CO 5	Identify gaps through literature review and propose creative solutions; organize and communicate technical findings effectively in written reports and oral presentations. (Cognitive Knowledge Level: Analyze)

Mapping of Course Outcomes with Program Outcomes

(Correlation levels: 3 = High, 2 = Medium, 1 = Low, blank = None)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	3	2	2	2	1		1			1
CO2	2	2	3	1	1	2	3	2	1	1	2
CO3					3	2	3	2	2	1	
CO4	1	1	1		2	1	1	2	3	3	3
CO5	2	3	2	2	2	1	1	1	2	3	2

SDG Relevance- Projects should align with United Nations Sustainable Development Goals (SDGs), such as SDG 9 (Industry, Innovation, and Infrastructure), SDG 7/13 (Clean Energy and Climate Action), SDG 11 (Sustainable Cities and Communities), and SDG 3/6 (Health and Clean Water), particularly in CO2 and problem definition.

Phase 1 Targets(unchanged core, minor emphasis on sustainability/ethics)

- Literature study/survey.
- Formulation of objectives.
- Formulation of hypothesis/design/methodology.
- Work plan and task allocation.
- Block-level design documentation.
- Seeking project funds.
- Preliminary analysis/ modeling/ simulation/ experiment/ design/ feasibility study/ prototype.
- Preparation of Phase 1 report.

Evaluation Guidelines & Rubrics

Total: 100 marks (Minimum to pass: 50).

- Project progress evaluation by guide : 30 Marks
- Interim evaluation by Evaluation Committee : 30 Marks
- Final evaluation by Evaluation Committee : 30 Marks
- Project Phase-I report (by Evaluation Committee) : 10 Marks

(The Evaluation Committee comprises HoD/Senior Faculty, Project Coordinator, and Supervisor.)

Evaluation by the Guide (30 Marks)

Sl. No.	Parameters	Marks	Poor (0–3 / 4–6)	Fair	Very Good (7–9)	Outstanding (10)
1-a	Topic identification, selection, formulation of objectives and/or literature survey (Group) [CO1/CO5]	15	Failed to identify relevant topic timely; no literature; no objectives clarity.	Identified topic but lacks originality/substance; limited references.	Good brainstorming; relevant topic; good literature; objectives mostly clear.	Highly relevant/innovative topic; excellent survey; clear, feasible objectives; strong societal/SDG relevance.
1-b	Project Planning, Scheduling, Resource/Tasks Identification & Allocation (Group) [CO4]	15	No evidence of planning/scheduling/budget; poor task allocation.	Some primary plan; incomplete schedules/journal.	Good planning; schedules prepared; better allocation.	Excellent planning; Gantt charts/modern tools; detailed budget; well-kept journal; clear individual tasks.

Interim Evaluation (30 Marks – Group mostly)

Sl. No.	Parameters	Marks	Poor (0–3 / 4–6)	Fair	Very Good (7–9)	Outstanding (10)
1-a	Topic identification, selection, formulation of objectives and/or literature survey (Group) [CO1/CO5]	15	Failed to identify relevant topic timely; no literature; no objectives clarity.	Identified topic but lacks originality/substance; limited references.	Good brainstorming; relevant topic; good literature; objectives mostly clear.	Highly relevant/innovative topic; excellent survey; clear, feasible objectives; strong societal/SDG relevance.
1-b	Project Planning, Scheduling, Resource/Tasks Identification & Allocation (Group) [CO4]	15	No evidence of planning/scheduling/budget; poor task allocation.	Some primary plan; incomplete schedules/journal.	Good planning; schedules prepared; better allocation.	Excellent planning; Gantt charts/modern tools; detailed budget; well-kept journal; clear individual tasks.

Final Evaluation (30 Marks – Mix of Group & Individual)

Sl. No.	Parameters	Marks	Poor	Fair	Very Good	Outstanding
1-c	Formulation of Design/ Methodology and Progress (Group) [CO1]	5	No evidence/ knowledge of design/ methodology; no progress.	Some knowledge; limited progress.	Comfortable with methods; good progress.	Clear/ well-defined methodology; excellent adherence/ progress.
1-d	Individual & Teamwork Leadership (Individual) [CO3]	10	Passive/ no interest.	Some interest (superficial).	Good interest/ responsibility/ team support.	Leadership; clear evidence of leading/ supporting team.
1-e	Preliminary Analysis/ Modeling/ Simulation / Experiment/ Design/ Feasibility (Group) [CO1/CO2]	10	No preliminary work.	Started but not thorough.	Good evidence; can improve.	Strong/ excellent work; poised for strong completion; shows innovation/ sustainability aspects.
1-f	Documentation & Presentation (Group/ Individual) [CO5]	5	Poor/ no documentation/ presentation.	Some but incomplete.	Good; needs improvement.	Professional (e.g., LaTeX); excellent structure/ presentation; clear communication of findings.

Project Phase-I Report (10 Marks)

Sl. No.	Parameter	Marks	Poor (0–3)	Fair (4–6)	Very Good (7–8)	Outstanding (9–10)
1-g	Report [CO5]	10	Shallow; no standard format; poor organization & references	Follows format partially; some organization issues; few references cited	Good organization; standard format; most references cited properly	Exceptionally organized; professional format; all references/ diagrams properly cited & presented; effective communication of technical content

General Guidelines

- Team size: Preferably 4 members (maximum 5 members), preferably from the same batch unless it is an interdisciplinary project.
- For the project, identify one base paper publication. It should be from IEEE, Elsevier, ACM journals, Science Citation Index (SCI) journals, or reputed Scopus-indexed journals. Conference papers are not permitted. Only recently published papers are allowed.
- For the literature survey, at least 10 publications are to be reviewed.
- Students are encouraged to seek funding from agencies such as KSCSTE, CERD, etc. Projects with societal relevance have a higher probability of attracting funding and recognition.
- Every student must maintain an individual project diary in the prescribed format. All project-related activities must be recorded with dates. The diary should be reviewed weekly by the project guide and signed (or approved via email in case of online mode). Project diaries may be collected periodically for evaluation.
- At the end of the seventh semester, a project report must be prepared in the prescribed format.
- Students should proceed with the project work with the objective of publishing at least one research paper during the eighth semester.

B24CS4L13	SEMINAR	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	4	1		2024

Preamble

The course 'Seminar' enables B.Tech graduates to search, read, comprehend, present, and report on academic/technical documents (e.g., peer-reviewed journals, conferences, books, theses, project reports). Students select a topic/paper in consultation with their guide, preferably linked to their final-year project area or emerging technologies with relevance to sustainable development. This fosters literature survey skills, presentation abilities, technical reporting, and awareness of broader societal/environmental impacts (aligned with SDGs like SDG 4: Quality Education, SDG 9: Industry/Innovation/Infrastructure, and SDG 13: Climate Action where applicable).

Course Objectives:

1. To conduct a literature survey in a selected engineering area of interest.
2. To understand and analyze academic/technical documents from peer-reviewed sources.
3. To prepare and deliver an effective presentation on the selected document/topic.
4. To prepare a comprehensive technical report with accurate references and consideration for sustainability and ethics where relevant.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Identify and select relevant academic documents from literature in their area of interest. (Cognitive Knowledge Level: Apply)
CO 2	Read, comprehend, and analyze an academic document/technical paper related to their area of interest. (Cognitive Knowledge Level: Analyze)
CO 3	Prepare a structured presentation on the selected academic document. (Cognitive Knowledge Level: Create)
CO 4	Deliver an effective presentation and engage in discussions/Q&A on the academic document. (Cognitive Knowledge Level: Apply)
CO 5	Prepare a well-structured technical report with proper references, formatting, and consideration for societal/sustainable aspects where applicable. (Cognitive Knowledge Level: Create)

Mapping of Course Outcomes with Program Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2	2	1	1		2	1				3
CO2	3	3	2	3		2	1				3
CO3	3	2	2		3			1		2	3
CO4	3				2			1		3	3
CO5	3	3	3	2	2	2	2	2		3	3

General Guidelines

- Department forms Internal Evaluation Committee (IEC): Academic coordinator (Chairperson), Seminar coordinator, and guide(s) as members. All IEC members must be present during presentations.
- IEC formation and guide allotment shall be completed within one week after previous semester examinations/end.
- Guide assists in topic/paper selection (current, broad-based; preferably linked to final-year project; consider SDG relevance where possible).
- Student selects topic/paper during semester break; finalize and submit to IEC in Week 1; IEC approval to be completed by Week 2.
- Accurate references from genuine peer-reviewed sources are required (to be verified in the report).
- Maintain a seminar diary for weekly progress monitoring by the guide.

Evaluation Pattern

Total marks: 100 (CIE only, minimum required to pass: 50)

1. Seminar Guide: 20 marks

- **Background Knowledge (10 marks):** Assessed by the guide based on the student's depth of understanding demonstrated during weekly interactions, discussions, and seminar diary entries.
- **Relevance of Topic/Paper Selected (10 marks):** Evaluated based on alignment with the discipline, currency, breadth, linkage to final-year project/SDGs (if applicable), and justification.

Rubric for Background Knowledge (10 marks)

Guide awards marks holistically or as an average of sub-criteria.

Level	Score	Descriptor
Excellent	9–10	Demonstrates exceptional grasp; explains concepts deeply, handles advanced questions confidently, shows critical insight beyond the paper.
Good	7–8	Solid understanding; explains key ideas clearly, answers most questions accurately, shows good analysis.
Satisfactory	5–6	Basic understanding; explains main points but with gaps, answers simple questions, limited depth.
Unsatisfactory	0–4	Poor/limited knowledge; struggles to explain basics, major misconceptions evident.

Rubric for Relevance of Topic/Paper (10 marks)

Level	Score	Descriptor
Excellent	9–10	Highly relevant, current/emerging topic; strong linkage to discipline/project/SDG; broad yet focused; peer-reviewed sources premium.
Good	7–8	Relevant and current; good project/SDG linkage; appropriate breadth.
Satisfactory	5–6	Moderately relevant; some linkage but narrow/specific or outdated.
Unsatisfactory	0–4	Low relevance; no clear linkage, outdated, or non-peer-reviewed dominant.

2. Seminar Coordinator: 20 marks

- **Seminar Diary (10 marks):** Assessed based on weekly progress entries, guide approvals, and regularity.
- **Attendance (10 marks)**

Rubric for Seminar Diary (10 marks)

Level	Score	Descriptor
Excellent	9–10	Complete, detailed weekly entries; clear progress milestones; insightful reflections; all approvals timely.
Good	7–8	Regular entries with good detail; most milestones tracked; approvals mostly on time.
Satisfactory	5–6	Basic entries; some gaps in detail/progress; approvals irregular.
Unsatisfactory	0–4	Incomplete/sparse entries; major gaps; few approvals.

3. Presentation: 40 marks (Awarded by IEC – all members)

Analytic rubric applied during the presentation session.

Criterion	Weight	Excellent (4) – Full marks scaled	Good (3)	Satisfactory (2)	Unsatisfactory (1)
Clarity of Presentation (Structure, flow, delivery, time management)	10	Exceptionally clear, logical flow, excellent delivery, perfect timing (within limits $\pm 10\%$).	Clear and well-structured, good delivery, minor timing issues.	Moderately clear, some disorganization, delivery fair, timing off.	Unclear, poor flow, mumbled delivery, major timing overrun/underrun.
Interactions / Q&A (Ability to answer questions, defend concepts, depth)	10	Confident, accurate, insightful answers; handles complex questions effectively.	Good answers to most questions; some depth shown.	Basic answers; struggles with follow-ups.	Poor/inaccurate answers; unable to defend key points.
Overall Participation (Involvement in peers' sessions – questions asked, feedback given)	10	Highly active; insightful questions/feedback in most sessions.	Active; contributes meaningfully in several sessions.	Minimal participation; occasional questions.	Little/no involvement in others' sessions.
Quality of Slides (Visual appeal, content organization, technical accuracy, references, minimal text)	10	Outstanding design; error-free, well-organized, excellent visuals/references.	Good design; minor errors, clear organization.	Adequate but cluttered/average visuals.	Poor design; many errors, overcrowded, missing references.

4. Report: 20 marks (Awarded by IEC)

Analytic rubric for the final technical report.

Criterion	Weight	Excellent (4) – Full scaled	Good (3)	Satisfactory (2)	Unsatisfactory (1)
Technical Content & Depth (Analysis quality, relevance, originality, understanding of paper)	8	In-depth critical analysis; original insights; excellent synthesis.	Good analysis; clear understanding; some insights.	Basic summary; limited analysis.	Superficial; major errors/gaps.
Overall Quality & Structure (Formatting, language, grammar, templates followed, logical flow)	5	Professional; flawless language/structure; adheres to template perfectly.	Good quality; minor errors; logical.	Average; noticeable errors; fair structure.	Poor formatting; many errors; disorganized.
Adequacy of References & Citations (Genuine peer-reviewed sources, proper style e.g. IEEE, in-text & list)	4	Extensive, high-quality sources; perfect citation style.	Adequate relevant sources; minor citation issues.	Basic sources; some citation errors.	Few/poor sources; major citation flaws/plagiarism risk.
Consideration for Sustainability/ Ethics/ Societal Impact (SDG linkage, ethical aspects, where applicable)	3	Strong, relevant discussion of impacts/SDGs/ethics.	Some discussion of broader impacts.	Minimal mention.	No consideration.

B24CS4T02	Viva-Voce	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	0	0	1	2024

Evaluation Scheme: Comprehensive Viva (CIE Only)

- **Evaluation Mode:** 100% Continuous Internal Evaluation (CIE) – No End Semester Examination (ESE).
Minimum pass: 50 marks out of 100.
- **Objective:** To evaluate the student's comprehensive understanding of core subjects studied up to 6th semester, ensuring retention, integration, and application of knowledge.
- **Scope:** Questions cover all core/program-specific courses from Semesters 1 to 6 (excluding open/minor electives unless specified). Includes:
 - Fundamental concepts and theories
 - Problem-solving and numerical analysis
 - Derivations and logical reasoning
 - Practical applications and case-based questions
 - Interconnections between subjects
- **Conduct:** Viva voce conducted by a panel consisting of an internal examiner and the HoD/senior faculty.
 - Duration: 15–30 minutes per student (individual)
 - May include follow-up questions
 - Blackboard explanations/derivations may be required

Evaluation Scheme & Rubrics (Total: 100 Marks)

The evaluation is holistic and panel-based. Marks are awarded individually based on performance during the viva.

Parameter	Marks	Poor (0–10)	Fair (11–20)	Good (21–30)	Very Good (31–40)	Excellent/ Outstanding (41–50)
Depth of Subject Knowledge (Core concepts, definitions, laws, theorems, derivations from all major core subjects up to S6)	50	Very limited recall; major gaps in fundamentals; unable to answer basic questions from >50% of core subjects.	Basic recall in some subjects; frequent prompting needed; gaps in key topics from multiple semesters.	Reasonable recall; answers most basic/medium questions correctly; minor gaps but can explain core ideas.	Strong grasp; handles medium- to advanced questions well across most subjects; explains derivations/applications confidently.	Exceptional depth; accurate, quick recall; integrates concepts across subjects (e.g., links circuits to signals, math to engineering applications); handles advanced/challenging questions fluently.
Understanding & Application (Ability to apply concepts, solve simple problems, explain real-world relevance, inter-connections)	20	No meaningful application; rote answers only; cannot relate concepts or solve basic problems.	Limited application; struggles with simple problems or real-world links.	Fair application; solves basic problems; some understanding of relevance.	Good application; solves medium-level problems; explains practical/industrial relevance.	Outstanding application; solves unseen problems creatively; clearly links concepts across domains and to societal/engineering contexts.
Clarity of Expression & Communication (Verbal clarity, logical flow, use of technical terms, confidence in answering)	15	Poor articulation; unclear/hesitant; frequent incomplete answers or mumbling.	Somewhat clear but disorganized; needs repetition/prompting often.	Clear and logical; most of the time; minor hesitations.	Fluent, well-structured answers; confident delivery; uses precise terminology.	Highly articulate, confident, and professional; excellent logical flow; engages panel effectively.

Parameter	Marks	Poor (0–10)	Fair (11–20)	Good (21–30)	Very Good (31–40)	Excellent/ Outstanding (41–50)
Overall Confidence, Attitude & Responsiveness (Body language, honesty in admitting unknowns, eagerness to learn/ explain further)	15	Nervous/ unresponsive; defensive or silent on difficult questions.	Somewhat confident but anxious; partial responses.	Generally confident; admits gaps reasonably.	Confident and composed; handles pressure well.	Highly confident, positive attitude; turns unknown questions into discussion points; demonstrates lifelong learning mindset.

Total: 100 Marks

Guidelines for Preparation & Conduct

- Students should revise all core theory and laboratory courses from Semesters 1 to 6.
- Focus on both fundamental concepts and advanced topics.
- The viva is typically scheduled in the last 1–2 weeks of the 7th semester.
- There is no written component; the evaluation is purely oral, but may include diagrams or derivations on the board.

B24CSH71	SURVEILLANCE VIDEO ANALYTICS (HONOURS)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5		

Preamble

The course introduces the concepts and techniques used in surveillance video analytics for intelligent monitoring and security applications. It covers video processing, object detection and tracking, deep learning methods, and behavior analysis in surveillance systems. After completing the course, students will be able to apply video analytics techniques to analyze surveillance data and design basic intelligent surveillance solutions.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO1	Explain the fundamentals of video and surveillance systems, including surveillance architectures, video structures, and compression standards. (Cognitive Knowledge Level: Understand)
CO2	Apply computer vision techniques such as preprocessing, motion detection, segmentation, and object tracking for surveillance applications. (Cognitive Knowledge Level: Apply)
CO3	Apply deep learning techniques for video analytics, including object detection, human counting, and activity recognition. (Cognitive Knowledge Level: Apply)
CO4	Explain event and behavior analysis methods used in surveillance systems, including crowd analysis, anomaly detection, and trajectory analysis. (Cognitive Knowledge Level: Understand)
CO5	Explain ethical, privacy, legal, and security considerations in AI-based surveillance systems. (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1			1				1	1	2
CO 2	3	3	2	3	3	1		1	1	1	3
CO 3	3	3	3	3	3	1		2	3	1	3
CO 4	3	3	2	3	3	3			1	1	3
CO 5	1	1	2	1	2	3	3	1	2	2	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	50	50	50
Apply	20	20	20
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (10 hours)

Fundamentals of Video & Surveillance Systems

Introduction to video surveillance, Analog vs digital surveillance systems, Video structure: frames, frame rate, resolution, Image vs video data characteristics, Surveillance architectures (CCTV, IP cameras), Challenges: illumination, occlusion, crowd density, Video compression standards (H.264, H.265), Camera calibration and basic geometry in surveillance.

MODULE 2 (11 hours)

Computer Vision for Surveillance

Image preprocessing (grayscale, filtering), Background subtraction, Motion detection techniques, Object detection basics, Feature extraction (edges, blobs, contours), Introduction to tracking concepts, Morphological operations in image processing, Region-based segmentation techniques.

MODULE 3 (8 hours)

Deep Learning in Video Analytics

Video representation and temporal features in video analytics, Deep learning object detection (YOLO/SSD concept), Human detection & counting, Action recognition basics, Pose estimation overview, Pretrained model usage, Transfer learning for video analytics applications, Introduction to spatio-temporal deep learning models (3D CNNs).

MODULE 4 (9 hours)

Event & Behavior Analysis

Object tracking (centroid, Kalman concept), Crowd analysis & density estimation, Anomaly detection in surveillance, Abnormal event detection, Traffic & public safety analytics, Smart city surveillance use cases, Trajectory analysis and pattern recognition, Rule-based vs learning-based behavior analysis.

MODULE 5 (7 hours)

Ethics & Privacy

Privacy issues in surveillance, Bias and fairness in AI vision, Legal & ethical frameworks, Responsible surveillance design, Dataset introduction (CCTV datasets), Data security and storage considerations in surveillance systems, Regulatory standards and compliance in surveillance technologies, Case Study: Object detection and tracking in surveillance videos.

Text Books

1. Richard Szeliski, Computer Vision: Algorithms and Applications, Springer.
2. Gary Bradski & Adrian Kaehler, Learning OpenCV: Computer Vision with the OpenCV Library, O'Reilly.
3. Saul Calderon & Manas Somaiya, Practical Deep Learning for Cloud, Mobile & Edge, O'Reilly.
4. Ian Goodfellow, Yoshua Bengio & Aaron Courville, Deep Learning, MIT Press.

Reference Books

1. Mark Nixon & Alberto Aguado, Feature Extraction & Image Processing for Computer Vision, Academic Press.
2. Wenhuan Zhange, Intelligent Video Surveillance: Systems and Technology, CRC Press.
3. Amarjot Singh & Devendra P. Dogra, Video Surveillance: The Vision Beyond Seeing, Springer.
4. D. Forsyth & J. Ponce, Computer Vision: A Modern Approach, Pearson.
5. Huimin Lu, Xiaokun Liu & Wei Zhao, Deep Learning for Intelligent Video Surveillance, Springer.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Fundamentals of Video & Surveillance Systems		10 hours
1.1	Introduction to video surveillance	1
1.2	Analog vs digital surveillance systems	1
1.3	Video structure: frames, frame rate, resolution	1
1.4	Image vs video data characteristics	1
1.5	Surveillance architectures (CCTV, IP cameras)	1
1.6	Challenges in surveillance: illumination, occlusion, crowd density	2
1.7	Video compression standards (H.264, H.265)	1

1.8	Camera calibration and basic geometry in surveillance	2
Module 2: Computer Vision for Surveillance		11 hours
2.1	Image preprocessing: grayscale conversion and filtering	1
2.2	Background subtraction techniques	1
2.3	Motion detection techniques	2
2.4	Object detection basics	2
2.5	Feature extraction: edges, blobs, and contours	1
2.6	Introduction to tracking concepts and Morphological operations in image processing	2
2.7	Region-based segmentation techniques	2
Module 3: Deep Learning in Video Analytics		8 hours
3.1	Video representation and temporal feature modelling in video analytics	1
3.2	Deep learning object detection: YOLO and SSD concepts	1
3.3	Human detection and crowd counting techniques	1
3.4	Action recognition fundamentals in surveillance videos	1
3.5	Pose estimation techniques	1
3.6	Use of pretrained deep learning models in video analytics	1
3.7	Transfer learning for surveillance video analytics applications	1
3.8	Spatio-temporal deep learning models: introduction to 3D CNNs	1
Module 4: Event & Behavior Analysis		9 hours
4.1	Object tracking techniques: centroid tracking and Kalman filter concept	2
4.2	Crowd analysis and density estimation methods	1
4.3	Anomaly detection in surveillance videos	2
4.4	Abnormal event detection and behavior understanding	1
4.5	Trajectory analysis and pattern recognition in surveillance data	2
4.6	Traffic and public safety analytics; smart city surveillance use cases; rule-based vs learning-based behavior analysis	1
Module 5: Ethics & Privacy		7 hours
5.1	Privacy issues in surveillance systems	1
5.2	Bias and fairness in AI-based vision systems	1
5.3	Legal and ethical frameworks in surveillance	1
5.4	Responsible surveillance system design principles	1

5.5	Dataset introduction: CCTV and surveillance datasets	1
5.6	Data security and storage considerations in surveillance systems	1
5.7	Regulatory standards and compliance; case study on object detection and tracking in surveillance videos	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Design a large-scale smart city video surveillance system for continuous urban monitoring. Explain the complete system architecture including camera types, network infrastructure, storage mechanisms, and video management strategies. Justify the choice of analog vs digital systems, resolutions, and frame rates for different monitoring scenarios.
2. Surveillance videos captured in crowded public environments often suffer from illumination variation, occlusion, and large data storage requirements. Analyze these challenges in detail and propose suitable technical solutions using video compression standards, camera placement strategies, and video parameter optimization.

Course Outcome 2 (CO 2):

1. A surveillance system deployed in a metro station must reliably detect moving objects under varying lighting conditions and dynamic backgrounds. Propose a complete computer vision pipeline including preprocessing, background modelling, motion detection, and segmentation techniques. Justify the choice of algorithms at each stage with respect to robustness and computational efficiency.
2. In a crowded surveillance scenario, accurate object detection and tracking depend heavily on effective feature extraction and region segmentation. Compare various feature extraction methods and segmentation approaches, and recommend a suitable combination for real-time surveillance applications. Explain how morphological operations improve detection accuracy in such environments.

Course Outcome 3 (CO 3):

1. A city surveillance department plans to deploy a deep learning system to automatically detect and count pedestrians at busy intersections. Design a suitable solution using deep learning object detection models and pretrained networks. Explain the model selection, training strategy, data requirements, and performance evaluation measures.

2. Develop an approach for recognizing suspicious human activities in surveillance videos using deep learning techniques. Describe how temporal video features, transfer learning, and spatio-temporal models can be used to improve recognition accuracy. Outline the implementation steps and practical challenges involved.

Course Outcome 4 (CO 4):

1. A smart traffic monitoring system must automatically track vehicles across multiple frames and identify unusual movement patterns. Propose a suitable approach using object tracking and trajectory analysis techniques. Explain how these methods help in detecting abnormal traffic events and improving public safety.
2. Design a surveillance solution for monitoring large public gatherings to prevent safety incidents. Describe how crowd density estimation, anomaly detection, and behavior analysis techniques can be integrated to identify potential risks. Discuss the practical challenges involved in real-time deployment.

Course Outcome 5 (CO 5):

1. (a) Explain the major privacy concerns associated with large-scale public surveillance systems and suggest suitable technical and policy measures to mitigate them.
(b) Discuss the importance of fairness and bias mitigation in AI-based surveillance systems. Propose practical steps to ensure responsible surveillance design.
2. (a) Describe the role of data security and storage management in surveillance systems handling large volumes of video data. Suggest best practices to prevent unauthorized access and misuse.
(b) Explain the need for regulatory standards and legal compliance in surveillance technologies. Illustrate your answer with a case study involving object detection and tracking in public spaces.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CSH71

Course Name: SURVEILLANCE VIDEO ANALYTICS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Differentiate between analog and digital surveillance systems.
2. List the major challenges faced in video surveillance under outdoor environments.
3. What is background subtraction and why is it important in surveillance video analysis?
4. Briefly explain the role of morphological operations in image processing.
5. What is transfer learning and how is it useful in video analytics applications?
6. Define spatio-temporal features in surveillance video analysis.
7. What is anomaly detection in surveillance systems? Give an example.
8. State the purpose of trajectory analysis in behavior monitoring.
9. List key privacy issues associated with AI-based surveillance systems.
10. What is the importance of regulatory compliance in surveillance technologies?

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the architecture of a modern IP-based video surveillance system and its components. (7)
(b) Compare analog and digital surveillance systems with respect to scalability, storage, and video quality. (7)

OR

12. (a) Describe the structure of video data and explain the significance of frame rate and resolution in surveillance. (6)
(b) Discuss major challenges in real-world surveillance environments and suggest suitable mitigation techniques. (8)
13. (a) Explain the steps involved in image preprocessing for surveillance video analysis. (7)
(b) Describe background subtraction and motion detection techniques used for moving object identification. (7)

OR

14. (a) Discuss various feature extraction techniques used in surveillance video processing. (6)
(b) Propose a method for detecting objects in crowded scenes using segmentation and morphological operations. (8)
15. (a) Explain deep learning-based object detection models suitable for surveillance applications. (7)
(b) Describe how pretrained models and transfer learning help in building efficient video analytics systems. (7)

OR

16. (a) Define temporal features in video analytics and explain their significance in activity recognition. (6)
(b) Design a deep learning approach for human action recognition in surveillance videos using spatio-temporal models. (8)
17. (a) Explain object tracking methods used in surveillance with suitable examples. (7)
(b) Describe trajectory analysis techniques and their role in behavior understanding. (7)

OR

18. (a) Discuss crowd density estimation methods for monitoring large public spaces. (6)
(b) Propose an anomaly detection framework for identifying abnormal events in surveillance videos. (8)
19. (a) Explain privacy concerns associated with large-scale surveillance systems. (7)
(b) Discuss methods to ensure fairness and responsible AI practices in surveillance applications. (7)

OR

20. (a) Describe data security and storage management practices in surveillance systems. (6)
(b) Discuss the importance of legal compliance and regulatory standards with a suitable surveillance case study. (8)

B24CSH72	COMPUTER AIDED DRUG DESIGN(HONOURS)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course introduces the fundamental principles and computational techniques used in modern drug discovery and development. It covers the concepts of molecular modelling, structure-based and ligand-based drug design, virtual screening, docking, QSAR, and molecular dynamics. Students gain conceptual understanding of how computational tools accelerate the drug discovery pipeline, along with exposure to widely used CADD tools, databases, and algorithms.

Prerequisites

Basic understanding of Chemistry and Biomolecules, Fundamentals of Algorithms and Data Structures, Basic knowledge of Python / computational tools.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the drug discovery process and fundamentals of CADD. (Cognitive Knowledge Level: Understand)
CO 2	Describe molecular modelling concepts including force fields, conformational analysis, and molecular dynamics. (Cognitive Knowledge Level: Understand)
CO 3	Apply structure-based drug design techniques such as homology modelling, docking, and virtual screening. (Cognitive Knowledge Level: Apply)
CO 4	Analyze ligand-based drug design approaches like QSAR, descriptor calculation, and statistical modelling. (Cognitive Knowledge Level: Analyze)
CO 5	Apply computational tools to design simple drug candidates and evaluate their ADMET properties. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1		1	1	1			1	1	3
CO 2	3	2		3	3				1		3
CO 3	2	3	3	3	3	1			1	1	3
CO 4	3	3	3	3	3	2			2	1	3
CO 5	3	3	3	3	3	3	3		3	2	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	50	50	50
Apply	20	20	20
Analyse	10	10	10
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (11 hours)

Introduction to Drug Discovery and CADD:

Overview of drug discovery pipeline: target identification, validation, lead discovery, optimization, preclinical & clinical studies; Need for computational approaches: time, cost & accuracy; Role and scope of CADD in pharmaceutical R&D; Types of CADD: Structure-Based Drug Design (SBDD) & Ligand-Based Drug Design (LBDD); Molecular representation: SMILES, InChI, fingerprints; Chemical databases: PubChem, ChEMBL, ZINC, PDB.

MODULE 2 (10 hours)

Molecular Modelling Fundamentals:

Basics of molecular mechanics & quantum mechanics; Force fields (MMFF, AMBER, CHARMM); Energy minimization & conformational analysis; Molecular dynamics: algorithms, trajectories, energy profiles, stability analysis and MD simulation parameters; Solvent models, implicit vs. explicit solvation; Applications of MD in drug design.

MODULE 3 (10 hours)

Structure-Based Drug Design (SBDD):

Protein structure basics: active site, binding pockets, motifs; Homology modelling: sequence alignment, template selection, model validation; Molecular docking principles: scoring functions, search algorithms; Virtual screening workflows; Pharmacophore modelling: hypothesis generation & validation; Case studies in SBDD.

MODULE 4 (7 hours)

Ligand-Based Drug Design (LBDD):

Quantitative Structure–Activity Relationship (QSAR); 1D, 2D & 3D descriptors; Feature selection & model building; Statistical tools for QSAR: regression, classification, validation metrics; Machine learning in drug discovery: SVM, RF, DL basics; ADMET prediction tools & in-silico screening.

MODULE 5 (7 hours)

Computational Drug Design Workflow and Applications:

Computational drug design workflow: target selection, ligand preparation, docking, scoring and ranking of compounds, Virtual screening techniques: structure-based and ligand-based screening methods for identifying potential drug candidates, Applications and case studies in computational drug discovery: integration of docking, QSAR, and ADMET in modern drug design pipelines.

Text Books

1. T.J. Ahamed, S.J. John Peters, Computational Chemistry and Molecular Modeling: Principles and Applications, Viva Books.
2. Andrew R. Leach, Molecular Modelling: Principles and Applications, Pearson Education.
3. K.B. Lipkowitz & D.B. Boyd, Reviews in Computational Chemistry, Wiley-VCH.

Reference Books

1. Gareth R. Jones & Peter Willett, Chemoinformatics: Applications of Chemical Information, Springer.
2. David C. Young, Computational Drug Design: A Guide for Computational and Medicinal Chemists, Wiley-Interscience.
3. Jürgen Bajorath (Ed.), Virtual Screening: Principles, Challenges, and Practical Guidelines, Wiley-Blackwell.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction to Drug Discovery and CADD		11 hours
1.1	Overview of drug discovery: introduction & historical context	1
1.2	Drug discovery pipeline: target identification and validation	1
1.3	Drug discovery pipeline: lead discovery and optimization	1
1.4	Preclinical and clinical studies in drug development	2
1.5	Need for computational approaches: time, cost, and accuracy advantages	1
1.6	Role and scope of Computer-Aided Drug Design (CADD) in pharmaceutical R&D	1
1.7	Types of CADD: Structure-Based Drug Design (SBDD) and Ligand-Based Drug Design (LBDD)	2
1.8	Molecular representations: SMILES, InChI, and molecular fingerprints	1
1.9	Chemical databases for drug discovery: PubChem, ChEMBL, ZINC, and Protein Data Bank	1
Module 2: Molecular Modelling Fundamentals		10 hours

2.1	Basics of molecular mechanics	1
2.2	Basics of quantum mechanics relevant to molecular modelling	1
2.3	Force fields: MMFF, AMBER, CHARMM and their components	2
2.4	Energy minimization techniques and conformational analysis	1
2.5	Molecular dynamics simulations: algorithms and trajectory analysis	2
2.6	Energy profiles, stability analysis and MD simulation parameters	1
2.7	Solvent models: implicit vs explicit solvation and applications of MD in drug design	2
Module 3: Structure-Based Drug Design (SBDD)		10 hours
3.1	Protein structure basics: active sites, binding pockets, and structural motifs	1
3.2	Homology modelling: sequence alignment and template selection	2
3.3	Homology modelling: model building and validation techniques	2
3.4	Molecular docking principles: search algorithms	1
3.5	Scoring functions and virtual screening workflows	2
3.6	Pharmacophore modelling and case studies in structure-based drug design	2
Module 4: Ligand-Based Drug Design (LBDD)		7 hours
4.1	Introduction to QSAR: concepts, workflow, and applications	1
4.2	Molecular descriptors: 1D physicochemical and constitutional descriptors	1
4.3	Molecular descriptors: 2D topological descriptors and fingerprints	1
4.4	Molecular descriptors: 3D geometric, steric, and electronic descriptors	1
4.5	Feature selection methods and QSAR model building techniques	1
4.6	Statistical tools for QSAR: regression, classification, and validation metrics	1
4.7	Machine learning in drug discovery and ADMET prediction for in-silico screening	1
Module 5: Computational Drug Design Workflow and Applications		7 hours

5.1	Overview of computational drug design workflow and target selection	1
5.2	Ligand preparation techniques and molecular structure optimization	1
5.3	Molecular docking workflow: scoring, ranking, and binding pose analysis	1
5.4	Virtual screening techniques: structure-based and ligand-based approaches	1
5.5	Post-screening analysis: hit identification and lead prioritization	1
5.6	Integration of docking, QSAR, and ADMET in drug design pipelines	1
5.7	Applications and case studies in computational drug discovery	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the stages of the drug discovery pipeline and describe how CADD accelerates each stage.
2. Illustrate the difference between Structure-Based Drug Design (SBDD) and Ligand-Based Drug Design (LBDD) with suitable examples.
3. Discuss the role of chemical databases such as PubChem, ChEMBL, and PDB in computational drug discovery.

Course Outcome 2 (CO 2):

1. Describe the components of a force field and explain how they contribute to molecular energy calculations.
2. Explain conformational analysis and discuss its significance in predicting stable molecular structures.
3. Illustrate the key steps of a molecular dynamics simulation and explain how MD helps in studying protein–ligand interactions.

Course Outcome 3 (CO 3):

1. Apply the steps of homology modelling to generate a 3D protein structure from a target sequence and describe how model validation is performed.

2. Explain the docking workflow and demonstrate how scoring functions are used to rank ligand poses in structure-based drug design.
3. Describe the process of virtual screening and identify factors influencing hit identification and prioritization.

Course Outcome 4 (CO 4):

1. Analyze the types of molecular descriptors used in QSAR modeling and explain their relevance in predicting biological activity.
2. Compare different statistical methods used for QSAR model development (e.g., regression, classification, machine learning).
3. Examine the model validation parameters such as R^2 , Q^2 , RMSE, and discuss their significance in assessing QSAR model performance.

Course Outcome 5 (CO 5):

1. Apply in-silico tools to design a potential lead molecule and explain the steps involved in ligand optimization.
2. Explain the use of ADMET prediction tools and evaluate how they assist in filtering out unsuitable drug candidates.
3. Demonstrate the workflow for constructing a simple drug design pipeline integrating molecular docking, QSAR, and ADMET screening.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CSH72

Course Name: COMPUTER AIDED DRUG DESIGN

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define Computer-Aided Drug Design (CADD). Explain its role in accelerating the drug discovery process.
2. Describe the differences between Structure-Based Drug Design (SBDD) and Ligand-Based Drug Design (LBDD).
3. What are force fields? Explain how they are used in molecular energy calculations.
4. Explain conformational analysis and discuss why it is important in drug design.
5. Describe the steps involved in homology modelling of a protein target.
6. What is molecular docking? Explain the role of scoring functions in ranking ligand poses.
7. Define QSAR. Explain the significance of molecular descriptors in QSAR modelling.
8. Compare regression-based and machine learning-based approaches for building QSAR models.
9. What is virtual screening? Discuss its importance in identifying potential lead compounds.
10. Explain ADMET prediction and its relevance in evaluating drug-likeness.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the stages of the drug discovery pipeline in detail. (8)
(b) Discuss how CADD reduces cost and time in modern drug discovery. (6)

OR

12. (a) Describe the different molecular representations (SMILES, InChI, fingerprints) used in CADD. (7)
(b) Explain the significance of target identification and validation in computational drug discovery. (7)
13. (a) Explain the components of a force field used in molecular mechanics. (9)
(b) Discuss the importance of energy minimization in drug design. (5)

OR

14. (a) Describe molecular dynamics simulation and explain how it helps in studying biomolecular behaviour. (8)
(b) Differentiate between implicit and explicit solvent models. (6)
15. (a) Explain the steps involved in homology modelling, including template selection, alignment, and validation. (7)
(b) Discuss the applications of homology models in drug discovery. (7)

OR

16. (a) Describe the molecular docking process with emphasis on search algorithms. (8)
(b) Explain the different types of scoring functions used in docking. (6)
17. (a) Explain the process of QSAR model development, including descriptor calculation. (9)
(b) Discuss the significance of model validation metrics such as R^2 , Q^2 , and RMSE. (5)

OR

18. (a) Describe 2D and 3D molecular descriptors with suitable examples. (7)
(b) Compare regression-based QSAR models with machine learning-based QSAR approaches. (7)
19. (a) Explain the workflow of virtual screening in drug discovery. (8)
(b) List and describe the key factors affecting hit prioritization in virtual screening. (6)

OR

20. (a) Discuss the importance of ADMET prediction in filtering drug candidates. (9)
(b) Explain how AI-based models improve the efficiency of drug design. (5)

B24CSH73	LOGIC FOR COMPUTER SCIENCE (HONOURS)	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		3	1	0	5	4	2024

Preamble

This course introduces the foundations of formal logic required for computer science. It covers propositional logic, predicate logic, deduction systems, resolution, temporal logic, modal logic, and program verification. The course emphasizes formal reasoning techniques used in designing and verifying computer systems. It also provides a strong theoretical foundation for areas such as artificial intelligence, formal methods, and software verification.

Prerequisites

Discrete Mathematics

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Understand mathematical and formal logic concepts used in computer science (Cognitive Knowledge Level : Understand)
CO 2	Apply propositional logic techniques to determine validity and satisfiability (Cognitive Knowledge Level : Apply)
CO 3	Use deduction systems and resolution techniques in propositional logic. (Cognitive Knowledge Level : Apply)
CO 4	Apply predicate logic and unification for automated reasoning. (Cognitive Knowledge Level : Apply)
CO 5	Assess the ethical responsibility of an engineer to choose formal program verification (Cognitive Knowledge Level : Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	2	2		1					1
CO 2	2	2	2	2		1					1
CO 3	2	2	2	2		1					1
CO 4	2	2	2	2		1					1
CO 5	2	2	2	2	2	2	1				2

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction and Propositional Logic :

Introductory Concepts: Mathematical Logic, Propositional Logic, Modal and Temporal logic, Program Verification. Propositional Logic: Formulae and interpretations, Equivalence, Satisfiability, Validity, Semantic Tableaux, Soundness and Completeness. The Hilbert Deductive System, Derived Rules, Theorems and operators, Soundness and Completeness, Consistency. **Case Study: Digital Circuit Verification**

MODULE 2 (8 hours)

Deduction in Propositional Logic and Binary Decision Diagrams :

Propositions and logical connectives, Logical equivalence, Conjunctive Normal Form (CNF), Clausal Form, Resolution Rule. Resolution Refutation. Binary Decision Diagrams (BDD): Definition, Reduced and Ordered BDD.

Case Study: Network Fault Diagnosis

MODULE 3 (10 hours)

Predicate Logic

Predicate Logic: Relations, predicates, formulae and interpretation, logical equivalence, semantic tableaux, soundness. The Hilbert deduction system for predicate logic. Functions, PCNF and clausal form, Herbrand model. Resolution in predicate logic: ground resolution, substitution, unification, general resolution.

MODULE 4 (7 hours)

Temporal Logic

Temporal logic: Syntax and semantics, models of time, linear time temporal logic, semantic tableaux. Deduction system of temporal logic. Program Verification: Need for verification, Framework for verification, Verification of sequential programs, deductive system, verification, synthesis.

MODULE 5 (6 hours)

Modal Logic

Modal Logic: Need for modal logic, Definition of Kripke frames and models; truth of modal formulae in a world; validity and satisfiability in modal logic. Case Study: Syntax and Semantics of K, axioms of K; soundness and completeness of system K.

Text Books

1. Arindhama Singh, Logics for Computer Science, Prentice Hall India, 2004.
2. Modechai Ben-Ari, Mathematical Logic for Computer Science, Springer, 3/e, 2012. Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, Pearson Education.
3. Logics for Computer and Data Sciences, and Artificial Intelligence — Lech T. Polkowski (2022–2024)

Reference Books

1. Logic in Computer Science — Hantao Zhang, Jian Zhang (2025)
2. An Introduction to Non-Classical Logic — Graham Priest (Second Edition)
3. First-Order Modal Logic — Melvin Fitting, Richard Mendelsohn (2023)

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		45 Hours
Module 1: Introduction and Propositional Logic		7 hours
1.1	Introduction to Mathematical Logic and its role in Computer Science	1
1.2	Overview of Propositional, Modal and Temporal Logic	1
1.3	Syntax of Propositional Logic: Formulae and Interpretations	1
1.4	Logical Equivalence, Satisfiability and Validity	1
1.5	Semantic Tableaux for Propositional Logic	1
1.6	Soundness and Completeness of Propositional Logic	1
1.7	Hilbert Deductive System: Axioms, Derived Rules, Theorems and Consistency	1
Module 2: Deduction in Propositional Logic and Binary Decision Diagrams		9 hours

2.1	Deduction Systems in Propositional Logic	1
2.2	Normal Forms: CNF and Clausal Form	1
2.3	Resolution Principle and Resolution Rule	1
2.4	Applications of Resolution in Propositional Logic	1
2.5	Introduction to Binary Decision Diagrams (BDD)	1
2.6	Ordered Binary Decision Diagrams (OBDD)	1
2.7	Reduced Ordered BDDs	1
2.8	Basic BDD Operators and Applications	1
2.9	Basic BDD Operators and Applications	1
Module 3: Predicate Logic		10 hours
3.1	Introduction to Predicate Logic: Relations and Predicates	1
3.2	Syntax and Semantics of Predicate Logic	1
3.3	Interpretations and Logical Equivalence	1
3.4	Semantic Tableaux for Predicate Logic	1
3.5	Soundness of Predicate Logic	1
3.6	Hilbert Deduction System for Predicate Logic	1
3.7	Functions, PCNF and Clausal Form	1
3.8	Herbrand Universe and Herbrand Model	1
3.9	Resolution in Predicate Logic: Ground Resolution and Substitution	1
3.10	Unification and General Resolution	1
Module 4: Temporal Logic and Program Verification		8 hours
4.1	Introduction to Temporal Logic and Motivation	1
4.2	Syntax and Semantics of Temporal Logic	1
4.3	Models of Time	1
4.4	Linear Time Temporal Logic (LTL) and Operators	1
4.5	Semantic Tableaux for Temporal Logic	1
4.6	Deduction System of Temporal Logic	1
4.7	Program Verification: Need, Framework and Verification of Sequential Programs	1
4.8	Program Verification: Need, Framework and Verification of Sequential Programs	1
Module 5: Modal Logic		9 hours
5.1	Need for Modal Logic and Applications in Computer Science	1

5.2	Syntax of Modal Logic	1
5.3	Kripke Frames and Kripke Models	1
5.4	Kripke Frames and Kripke Models	1
5.5	Truth of Modal Formulae, Validity and Satisfiability	1
5.6	Modal System K: Syntax, Semantics and Axioms	1
5.7	Modal System K: Syntax, Semantics and Axioms	1
5.8	Soundness and Completeness of System K	1
5.9	Soundness and Completeness of System K	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain semantic tableaux for propositional logic with an example.
2. Discuss the Hilbert deductive system and derived rules.
3. Explain propositional logic in detail, including formulae, interpretations, equivalence, satisfiability, validity, soundness, and completeness.

Course Outcome 2 (CO 2):

1. Explain the resolution principle in propositional logic.
2. Explain resolution in propositional logic and discuss Binary Decision Diagrams with suitable examples.
3. Describe ordered and reduced BDDs.

Course Outcome 3 (CO 3):

1. Explain PCNF and clausal form in predicate logic.
2. Describe ground resolution with an example.
3. Explain resolution in predicate logic, including substitution, unification, and Herbrand model.

Course Outcome 4 (CO 4):

1. Explain the syntax and semantics of temporal logic.

2. Describe the need for program verification.
3. Explain temporal logic and discuss its role in program verification of sequential programs.

Course Outcome 5 (CO 5):

1. Explain the concept of optical flow and describe the difference between global and local optical flow methods (Horn–Schunck vs Lucas–Kanade).



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, DECEMBER 2027

Course Code: B24CSH73

Course Name: LOGIC FOR COMPUTER SCIENCE

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define mathematical logic.
2. What is a well-formed formula in propositional logic?
3. Define satisfiability.
4. What is semantic tableau?
5. What is conjunctive normal form (CNF)?
6. Define Binary Decision Diagram (BDD).
7. What is a predicate?
8. Define unification.
9. What is linear time temporal logic (LTL)?
10. Define Kripke frame.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain semantic tableaux for propositional logic and discuss soundness and completeness. (7)
(b) Describe the Hilbert deductive system for propositional logic, including axioms, derived rules, and consistency. (7)

OR

12. (a) Explain formulae and interpretations in propositional logic with suitable illustrations. (7)
(b) Explain the concept of consistency in a logical system and its significance. (7)
13. (a) Explain conjunctive normal form (CNF) and clausal form with suitable examples. (7)
(b) Describe the resolution principle in propositional logic and illustrate it with an example. (7)

OR

14. (a) Describe reduced ordered Binary Decision Diagrams (ROBDD) and explain why reduction is important. (7)
(b) Write short notes on normal forms in propositional logic. (7)
15. (a) Describe relations and predicates and explain how interpretations are defined in predicate logic. (7)
(b) What is prenex conjunctive normal form (PCNF)? Explain the steps to convert a predicate logic formula into PCNF. (7)

OR

16. (a) Explain clausal form and its importance in predicate logic resolution. (7)
(b) Discuss substitution and unification in predicate logic with suitable examples. (7)
17. (a) Explain linear time temporal logic (LTL) and its temporal operators. (7)
(b) Explain the verification of sequential programs using deductive methods. (7)

OR

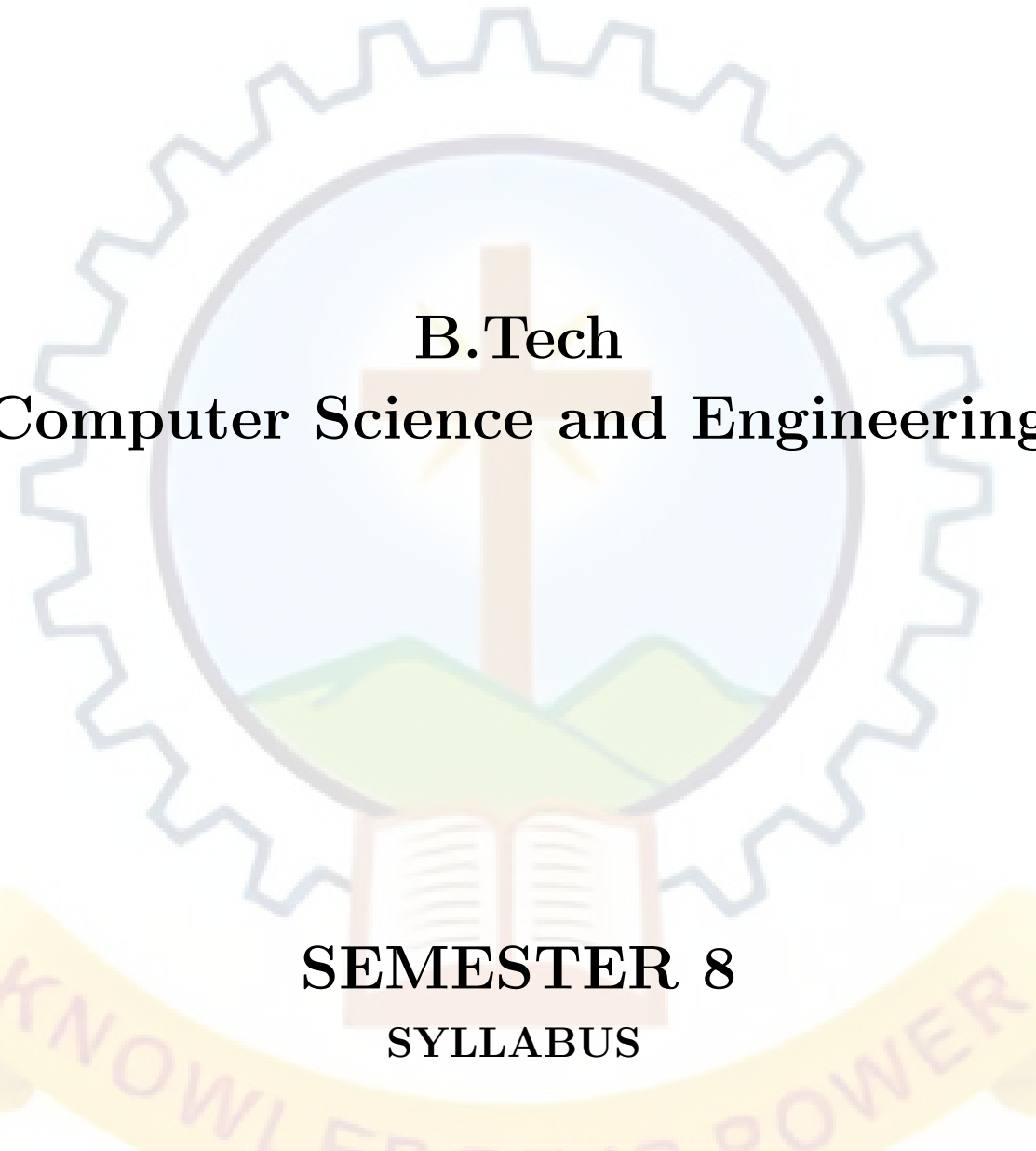
18. (a) Explain the deduction system of temporal logic. (7)
(b) Explain the verification of sequential programs using deductive methods. (7)
19. (a) Explain Kripke frames and Kripke models with suitable examples. (7)
(b) Write short notes on applications of modal logic in program specification and verification. (7)

OR

20. (a) Explain the need for modal logic and discuss its applications in computer science. (7)
(b) Compare propositional logic and modal logic highlighting their limitations and advantages. (7)

MAR ATHANASIOUS COLLEGE OF ENGINEERING

Government Aided, Autonomous Institution
Kothamangalam, Kerala, India

The logo of Mar Athanasius College of Engineering is a circular emblem. It features a large gear-like outer border. Inside the circle, there is a light blue background with a central orange cross. Below the cross, there are green hills. At the bottom of the circle, there is an open book. A yellow banner with the text 'KNOWLEDGE IS POWER' in purple capital letters curves around the bottom of the emblem.

B.Tech Computer Science and Engineering

SEMESTER 8 SYLLABUS

SEMESTER 8

SLOT	COURSE NO.	COURSES	L-T-P-S	HOURS	CREDIT
A,B,C	B24MC400x	INTERNSHIP & MOOC COURSES (3 NUMBERS)	0-0-0-0	0	9
OR					
A	B24CS4P5x	PROGRAMME ELECTIVE V	2-1-0-4	3	3
B	B24CS4P6x	PROGRAMME ELECTIVE VI	2-1-0-4	3	3
C	B24CS4G3x	OPEN ELECTIVE III	2-1-0-4	3	3
AND					
H	B24CS4L14	PROJECT PHASE II	0-0-12-3	12	6
M	B24CSML1	MINOR PROJECT	0-0-3-12	3	6
N	B24CSHL1	HONOURS PROJECT	0-0-6-9	6	6
TOTAL				30	15

PROGRAMME ELECTIVE V

COURSE NO.	COURSES
B24CS4P51	BUSINESS ANALYTICS (Bucket-I)
B24CS4P52	CLOUD SECURITY (Bucket-II)
B24CS4P53	BIOMETRICS (General-I)
B24CS4P54	PROGRAMMING IN R (General-II)

PROGRAMME ELECTIVE VI

COURSE NO.	COURSES
B24CS4P61	SOFTWARE TESTING PLATFORMS AND TOOLS (Bucket-I)
B24CS4P62	BLOCKCHAIN TECHNOLOGIES (Bucket-II)
B24CS4P63	MOBILE COMPUTING (General-I)
B24CS4P64	SOFT COMPUTING (General-II)

OPEN ELECTIVE III

COURSE NO.	COURSES
B24CS4G31	INTRODUCTION TO OPERATING SYSTEMS
B24CS4G32	BASICS OF CLIENT SERVER COMPUTING
B24CS4G33	INTRODUCTION TO DISTRIBUTED COMPUTING

MINOR

COURSE NO.	COURSES
B24CSML1	MINOR PROJECT

HONOURS

COURSE NO.	COURSES
B24CSHL1	HONOURS PROJECT

B24CS4P51	BUSINESS ANALYTICS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course introduces the fundamental concepts of Business Analytics, focusing on how data can be used to support effective business decision-making. It familiarizes students with different types of analytics—descriptive, predictive, and prescriptive—and the processes involved in collecting, analyzing, and interpreting data. The course also highlights the use of analytical tools and techniques in solving real-world business problems, helping students understand how data-driven insights can improve organizational performance.

Prerequisites

Basic knowledge of information systems, statistics, and business environment

Course Outcomes

After the completion of the course the student will be able to:

CO1	Understand the fundamentals of business intelligence, analytics, and data science. (Cognitive Knowledge Level: Understand)
CO2	Apply statistical methods and analytical models for real-life decision making. (Cognitive Knowledge Level: Apply)
CO3	Explain data warehousing concepts and apply data mining techniques for business analytics. (Cognitive Knowledge Level: Apply)
CO4	Describe text analytics and apply basic text-mining approaches. (Cognitive Knowledge Level: Apply)
CO5	Understand business performance management, reporting, and the impact of emerging technologies, legal, and ethical issues in analytics. (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2		1					1		
CO 2	3	3	1	3	2				1		
CO 3	2	3	2	3	2				1		1
CO 4	2	3	2	3	3				2		2
CO 5	3	3	3	3	3	3	3		3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	20	30
Understand	50	50	50
Apply	20	30	20
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Introduction to Business Analytics

Definition, scope and importance in modern organizations. Types of Analytics – Descriptive, Predictive and Prescriptive - meaning, characteristics, differences, and examples. Role of analytics in decision-making- Business Analytics lifecycle - Problem definition, data collection, data preparation, model building, evaluation, deployment, and monitoring. Application of Business Analytics in industries- case studies.

MODULE 2 (10 hours)

Data Fundamentals for Business Analytics

Types of data (structured and unstructured). Sources of Data and Data collection methods. Data quality and pre-processing for analytics.

Statistical Modelling for Business Analytics

Descriptive Statistics- Mean, Median, Variance. Probability and distributions. Correlation, regression.(Linear and Logistic) and Time Series forecasting.

MODULE 3 (7 hours)

Business Performance Management

Business Performance Management – concept, objectives, and importance. Key Performance Indicators (KPIs) and Metrics Design – definition, characteristics of effective KPIs, types of KPIs, guidelines for designing meaningful performance metrics. Analytics in Business Support Functions – application of analytics in Sales and Marketing , Human Resources, Financial Analytics, and Production and Operations.

MODULE 4 (7 hours)

Business Intelligence

Definition, Need, and Evolution of Business Intelligence System and its components.

BI Tools overview- Power BI, Tableau, Looker.

Excel for Business Analytics- Advanced Excel functions (VLOOKUP, INDEX-MATCH, Pivot Tables).

Web Analytics overview.

MODULE 5 (6 hours)

Emerging Trends and Ethical Considerations in Business Analytics

Recent Trends, Privacy and Managerial Considerations in Analytics

Use of Internet of Things for Business Analytics - Cloud Computing and Business Analytics.

Location Based Analytics for Organisations - Issues of Legality, Privacy and Ethics - Impacts of Analytics in Organisations.

Text Books

1. R. Sharda, D. Delen, and E. Turban, "Business Intelligence, Analytics, and Data Science: A Managerial Perspective", Pearson, 4th edition, 2018.
2. S. Christian Albright and Wayne L. Winston, "Business Analytics: Data Analysis and Decision Making", Cengage Learning, 2011.

Reference Books

1. R. N. Prasad and S. Acharya, "Fundamentals of Business Analytics", Wiley, 2nd Edition, 2016
2. J. R. Evans, "Business Analytics", Pearson, 3rd Edition, 2019.
3. A. Maheshwari, "Data Analytics", McGraw Hill Education, 1st Edition, 2017
4. Business Analytics for Managers: Taking Business Intelligence Beyond Reporting, 2nd Edition, Wiley, 2016

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to Business Analytics		6 hours
1.1	Introduction to Business Analytics – Definition	1
1.2	Types of Analytics – Descriptive, Predictive, Prescriptive	1
1.3	Role of Analytics in Decision Making	1
1.4	Business Analytics Lifecycle	1
1.5	Applications of Business Analytics in Industries	1
1.6	Case Studies in Business Analytics	1
Module 2: Data Fundamentals and Statistical Modelling		10 hours
2.1	Types of Data – Structured and Unstructured	1
2.2	Sources of Data and Data Collection Methods	1
2.3	Data Quality and Data Pre-processing	1
2.4	Descriptive Statistics – Mean and Median	1
2.5	Descriptive Statistics – Variance	1
2.6	Probability and Probability Distributions	1

2.7	Correlation Analysis	1
2.8	Linear Regression	1
2.9	Logistic Regression	1
2.10	Time Series Forecasting	1
Module 3: Business Performance Management		7 hours
3.1	Introduction to Business Performance Management	1
3.2	Key Performance Indicators (KPIs)	1
3.3	Metrics Design	1
3.4	Analytics in Sales and Marketing	1
3.5	Analytics in Human Resources and Financial Analytics	2
3.6	Analytics in Production and Operations	1
Module 4: Business Intelligence and Tools		7 hours
4.1	Business Intelligence – Definition, Need and Evolution	1
4.2	Components of Business Intelligence Systems	1
4.3	BI Tools Overview – Power BI	1
4.4	BI Tools Overview – Tableau and Looker	1
4.5	Excel for Business Analytics – VLOOKUP and INDEX-MATCH	1
4.6	Excel for Business Analytics – Pivot Tables	1
4.7	Web Analytics Overview	1
Module 5: Emerging Trends and Ethical Considerations		6 hours
5.1	Recent Trends in Business Analytics	1
5.2	Privacy and Managerial Considerations	1
5.3	Internet of Things for Business Analytics	1
5.4	Cloud Computing and Business Analytics	1
5.5	Location-Based Analytics for Organisations	1
5.6	Legal, Ethical Issues and Impact of Analytics in Organisations	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Differentiate between descriptive, predictive, and prescriptive analytics.
2. Explain the role of business analytics in decision making.

3. What is the business analytics lifecycle? Briefly explain its stages.

Course Outcome 2 (CO2):

1. What are the different types of data? Explain structured and unstructured data.
2. What is data preprocessing? Why is it important in analytics?
3. Explain correlation and regression with suitable examples.
4. Write a short note on time series forecasting.

Course Outcome 3 (CO3):

1. What are Key Performance Indicators (KPIs)? Give examples.
2. Explain the concept of metrics design in business performance management.
3. Describe the role of analytics in sales and marketing.
4. How is analytics used in financial and operations management?

Course Outcome 4 (CO4):

1. Define Business Intelligence and explain its need.
2. List and describe the components of a Business Intelligence system.
3. Compare Power BI and Tableau as business analytics tools.
4. Explain VLOOKUP and Pivot Tables in Excel with their applications.
5. What is web analytics? Mention its applications.

Course Outcome 5 (CO5):

1. What are the recent trends in business analytics?
2. Explain the role of IoT in business analytics.
3. How does cloud computing support business analytics?
4. What are the major privacy and ethical issues in analytics?
5. Discuss the impact of analytics in organisations.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

EIGHTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2028

Course Code: B24CS4P51

Course Name: BUSINESS ANALYTICS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define Business Analytics. How is it different from Business Intelligence?
2. Differentiate between descriptive, predictive and prescriptive analytics.
3. What are the measures of central tendency?
4. Define variance and its importance.
5. What is data preprocessing? Why is it required?
6. Distinguish between structured and unstructured data.
7. What are Key Performance Indicators (KPIs)?
8. List the components of a Business Intelligence system.
9. What is web analytics?
10. State any two privacy issues in business analytics.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. Explain the business analytics lifecycle. Discuss the role of analytics in decision making with examples. (14)

OR

12. Discuss the applications of business analytics in various industries with suitable case studies. (14)

13. (a) Explain data collection methods and data preprocessing steps. (7)
(b) Explain correlation and linear regression with examples. (7)

OR

14. (a) Explain probability distributions with examples. (7)
(b) Describe logistic regression and time series forecasting. (7)
15. (a) Explain Key Performance Indicators (KPIs) and metrics design. (7)
(b) Describe analytics in sales and marketing. (7)

OR

16. Explain the role of analytics in business functions such as HR, finance, and operations. (14)
17. (a) Define Business Intelligence and explain its components. (7)
(b) Explain the use of Excel functions (VLOOKUP, INDEX-MATCH, Pivot Tables) in analytics. (7)

OR

18. Discuss BI tools such as Power BI, Tableau, and Looker. Also explain web analytics. (14)
19. (a) Explain the role of IoT in business analytics. (7)
(b) Discuss cloud computing in business analytics. (7)

OR

20. Discuss privacy, legal, and ethical issues in business analytics. Also explain the impact of analytics in organisations. (14)

B24CS4P52	CLOUD SECURITY	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4		2024

Preamble

This course provides a comprehensive understanding of cloud security principles, architectures, and practices. It covers security challenges in cloud environments, including infrastructure, applications, and data protection. The course also introduces governance, compliance, and emerging cloud security trends.

Prerequisites

Basic knowledge of Computer Networks and Operating Systems

Course Outcomes

CO 1	Explain cloud security fundamentals and threat landscape.(Cognitive Knowledge Level : Understand)
CO 2	Analyze security challenges in virtualization and cloud infrastructure.(Cognitive Knowledge Level : Analyze)
CO 3	Apply identity, access control, and data protection mechanisms. (Cognitive Knowledge Level : Apply)
CO 4	Evaluate security practices in cloud applications and operations.(Cognitive Knowledge Level : Analyze)
CO 5	Assess governance, compliance, and emerging cloud security challenges.(Cognitive Knowledge Level : Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	1	1	2	1	2			2	1	2
CO 2	2	2	1	1	1	1			1		1
CO 3	2	2	2	1	1	1	2		1	1	
CO 4	2	2	2	2	3	1		2	2	2	2
CO 5	2	2	2	2	1	3	3	1	2	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (% Marks)	Test 2 (% Marks)	
Remember	15	10	10
Understand	45	25	30
Apply	25	35	35
Analyze	15	30	25
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Cloud Security Foundations and Threat Landscape:

Cloud security fundamentals, shared responsibility model, cloud architectures, threat landscape, cloud-specific attacks including data breaches, account hijacking, misconfiguration and insecure APIs.

MODULE 2 (7 hours)

Infrastructure and Virtualization Security:

Virtualization security, hypervisor threats, multi-tenancy, container security, secure networking (VPC, segmentation, firewalls), DDoS attacks and mitigation.

MODULE 3 (7 hours)

Identity, Access and Data Security:

IAM, privileged access, encryption (data at rest, transit, use), key management, data lifecycle protection.

MODULE 4 (8 hours)

Cloud Application and Operational Security:

Cloud-native security, API security, DevSecOps, monitoring, intrusion detection, vulnerability assessment, incident response.

MODULE 5 (7 hours)

Governance, Compliance and Emerging Security:

Risk management, NIST, CSA, ISO 27001, compliance, cloud forensics, serverless and edge security.

Text Books

1. Tim Mather et al., *Cloud Security and Privacy*, O'Reilly.
2. Thomas Erl et al., *Cloud Computing: Concepts, Technology & Architecture*.

Reference Books

1. Vic Winkler, *Securing the Cloud*, Elsevier.
2. John R. Vacca, *Cloud Computing Security*, CRC Press.
3. NIST SP 800-145, CSA Security Guidance.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	Hours
Total Hours		36 hours
Module 1		7 hours
1.1	Fundamentals	1
1.2	Responsibility model & architecture	1
1.3	Responsibility model & architecture	1
1.4	Threat landscape	1
1.5	Threat landscape	1
1.6	Cloud attacks	1
1.7	Cloud attacks	1
Module 2		7 hours
2.1	Virtualization security	1
2.2	Virtualization security	1
2.3	Multi-tenancy	1
2.4	Container security	1
2.5	Networking security	1
2.6	Networking security	1
2.7	DDoS mitigation	1
Module 3		7 hours
3.1	IAM	1
3.2	IAM	1
3.3	Privileged access	1
3.4	Encryption	1
3.5	Encryption	1
3.6	Key management	1
3.7	Data lifecycle	1
Module 4		8 hours
4.1	Cloud-native security	1
4.2	API security	1
4.3	DevSecOps	1
4.4	DevSecOps	1
4.5	Monitoring & IDS	1

4.6	Monitoring & IDS	1
4.7	Vulnerability management	1
4.8	Incident response	1
Module 5		7 hours
5.1	Risk management	1
5.2	Risk management	1
5.3	Standards	1
5.4	Standards	1
5.5	Compliance	1
5.6	Forensics	1
5.7	Emerging security	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO 1):

1. Explain the shared responsibility model in cloud computing. Discuss how security responsibilities differ between cloud providers and customers in IaaS, PaaS and SaaS environments.
2. Describe the cloud threat landscape and explain the causes, impacts and mitigation strategies of data breaches, account hijacking and insecure APIs.

Course Outcome 2 (CO 2):

1. Analyze the security challenges associated with virtualization, including hypervisor attacks and multi-tenancy risks. Suggest suitable mitigation techniques.
2. Examine how Virtual Private Clouds (VPCs), network segmentation and firewalls contribute to cloud infrastructure security. Explain their role in mitigating DDoS attacks.

Course Outcome 3 (CO 3):

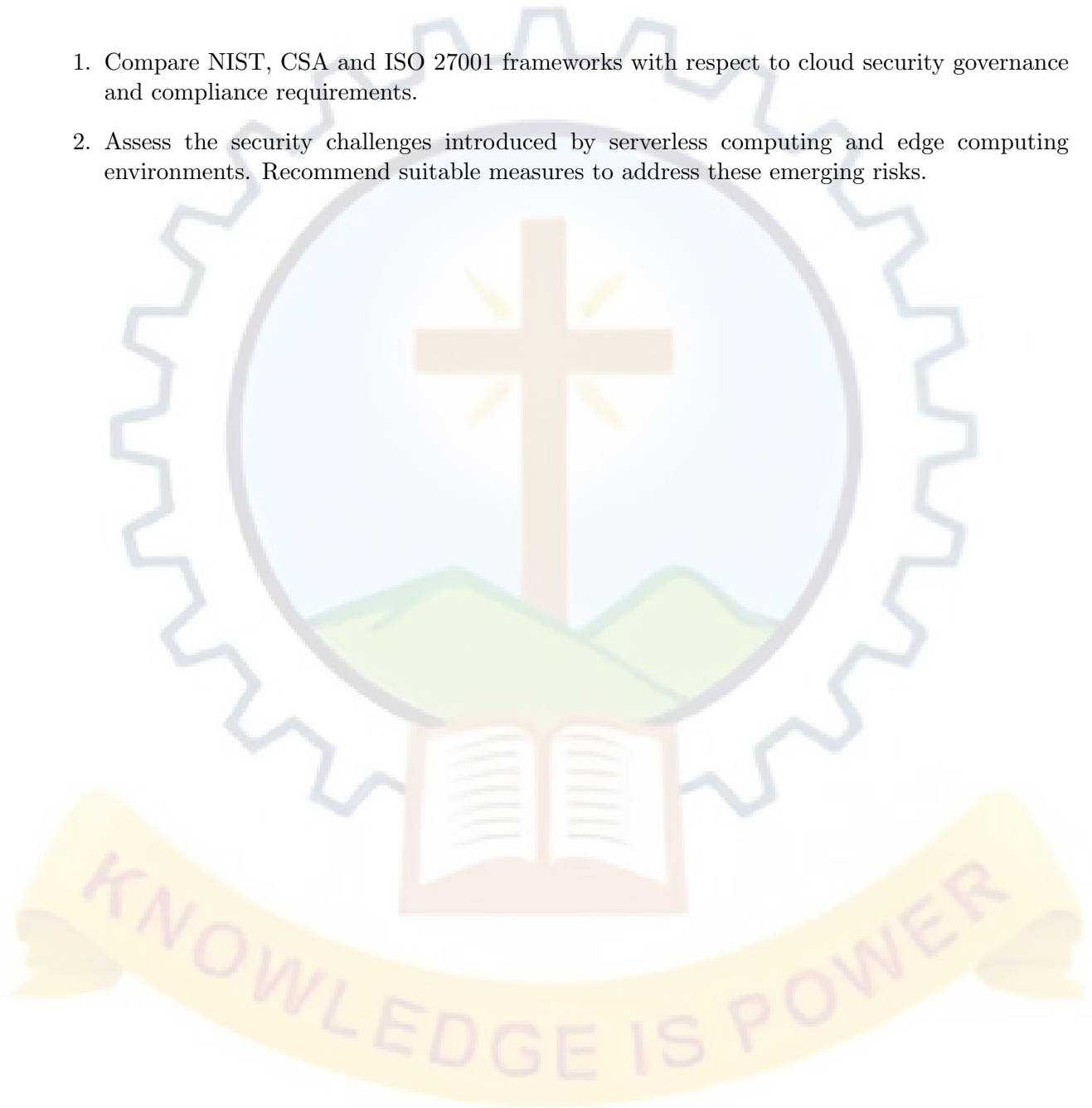
1. Design an Identity and Access Management (IAM) framework for an enterprise cloud environment incorporating role-based access control and privileged access management.
2. A healthcare organization stores confidential patient records in the cloud. Demonstrate how encryption and key management techniques can be applied to protect data at rest, in transit and in use.

Course Outcome 4 (CO 4):

1. Evaluate the effectiveness of DevSecOps practices in improving cloud application security. Explain how security can be integrated throughout the software development lifecycle.
2. Assess the importance of monitoring, intrusion detection, vulnerability assessment and incident response in maintaining cloud operational security.

Course Outcome 5 (CO 5):

1. Compare NIST, CSA and ISO 27001 frameworks with respect to cloud security governance and compliance requirements.
2. Assess the security challenges introduced by serverless computing and edge computing environments. Recommend suitable measures to address these emerging risks.



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

EIGHTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2028

Course Code: B24CS4P52

Course Name: CLOUD SECURITY

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define the shared responsibility model in cloud computing.
2. Differentiate between public cloud and private cloud deployment models.
3. What is a hypervisor? Mention any two security threats associated with hypervisors.
4. Explain the concept of multi-tenancy in cloud environments.
5. Define Identity and Access Management (IAM). State its importance in cloud security.
6. Differentiate between data encryption at rest and data encryption in transit.
7. What is API security? List any three common API vulnerabilities.
8. Explain the role of intrusion detection systems in cloud environments.
9. List the major objectives of cloud risk management.
10. What is serverless computing? Mention one security challenge associated with it.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain cloud security fundamentals and discuss the shared responsibility model across IaaS, PaaS and SaaS environments. (7)

- (b) Describe cloud-specific attacks such as data breaches, account hijacking, misconfiguration and insecure APIs. Suggest suitable mitigation strategies. (7)

OR

12. Explain cloud architectures and deployment models. Compare public, private and hybrid clouds from a security perspective. (14)
13. Analyze virtualization security issues including hypervisor threats and multi-tenancy risks. Discuss security controls to mitigate these threats. (14)

OR

14. (a) Explain the role of VPCs, network segmentation and cloud firewalls in securing cloud infrastructure. (7)
- (b) Describe Distributed Denial of Service (DDoS) attacks and discuss commonly used mitigation mechanisms in cloud environments. (7)
15. (a) Explain Identity and Access Management (IAM) and discuss role-based access control and privileged access management. (7)
- (b) Describe encryption mechanisms for protecting data at rest, data in transit and data in use. Explain the role of key management systems. (7)

OR

16. A multinational organization is migrating its critical business applications to the cloud.
- (a) Design a secure IAM strategy for managing employees, contractors and administrators. (6)
- (b) Recommend suitable encryption and key management practices to protect sensitive organizational data throughout its lifecycle. (8)
17. (a) Explain cloud-native security and discuss the principles of DevSecOps. (6)
- (b) Evaluate the importance of monitoring, intrusion detection and vulnerability assessment in cloud operational security. (8)

OR

18. A cloud-hosted e-commerce platform experiences unusual traffic spikes and unauthorized API access attempts.
- (a) Discuss the API security measures that should be implemented to protect the application. (6)
- (b) Develop an incident response plan including detection, containment, eradication and recovery phases. (8)
19. Compare NIST, CSA and ISO 27001 frameworks with respect to cloud governance, risk management and compliance. Discuss their significance in organizational cloud security programs. (14)

OR

20. (a) Explain the process of cloud forensics and identify the challenges involved in conducting forensic investigations in cloud environments. (6)
- (b) Assess the security challenges associated with serverless computing and edge computing. Recommend appropriate controls to address these emerging threats. (8)

B24CS4P53	BIOMETRICS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course serves to introduce the foundational principles, technologies, and applications of biometric security systems. The contents comprehensively cover operational concepts, key modalities like fingerprint, facial, and iris recognition, and critical considerations for implementation including privacy and standards. Studying this course provides essential knowledge for future careers in the rapidly expanding fields of cybersecurity, digital identity management, and secure system design.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO1	Understand the core principles, operational steps, and derived accuracy metrics (FMR, FNMR, FTE) of major biometric modalities, including the underlying natural science principles (e.g., physiology, optics) and engineering fundamentals of biometric systems. (Cognitive Knowledge Level: Understand)
CO2	Design a comparative matrix to evaluate the strengths, weaknesses, and applicability of different biometric technologies for specific security scenarios, applying mathematical and statistical concepts (e.g., error rates, thresholds) with appropriate engineering practice. (Cognitive Knowledge Level: Apply)
CO3	Use statistical error rates to determine the most suitable biometric system configuration for a given security requirement, while considering resource efficiency, whole-life cost, and reproducibility in system deployment. (Cognitive Knowledge Level: Apply)
CO4	Compare the privacy factors and ethical considerations associated with different biometric technologies, and evaluate the societal and environmental consequences (e.g., surveillance, data protection, bias) of biometric deployments. (Cognitive Knowledge Level: Apply)
CO5	Select and justify appropriate biometric technologies for a layered security solution to address a defined network security problem, demonstrating engagement with current research literature and inclusive team practices in biometric project development. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1	1	1	1						1
CO 2	2	3	2	3	3	1	3		2	2	3
CO 3	2	3	2	3	3	1			2	2	3
CO 4	1	2	2	3	3	3	3		2		1
CO 5	1	3	3	3	3	3	3	3	3	2	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Foundations of Biometric Systems

Introduction – Benefits of biometric security – Verification and identification – Basic working of biometric matching – Accuracy – False match rate – False non-match rate – Failure to enroll rate – Derived metrics – Layered biometric solutions.

MODULE 2 (7 hours)

Facial Recognition and Iris Recognition Technology

Facial Scan - Features –Components – Operation (Steps) – Competing facial Scan technologies – Strength and weakness. Introduction to iris recognition, Iris features and uniqueness, Components of iris recognition system, Steps in iris recognition process, Feature extraction techniques, Iris matching methods - Hamming distance matching, Correlation based matching, Euclidean distance matching, Competing iris technologies, Strengths and weaknesses

MODULE 3 (7 hours)

Fingerprint Technology

Finger scan – Features –Components – Operation (Steps)-Image Acquisition Image Processing, Location of Distinctive Characteristics, Template Creation Template Matching – Competing finger Scan technologies –optical, silicon, Ultrasound, Feature Extraction Methods. Strength and weakness. Types of algorithms used for interpretation.

MODULE 4 (8 hours)

Voice Recognition and Multimodal Biometric Technologies

Voice Scan - Features – Components – Operation (Steps) – Competing voice Scan (facial) technologies – Strength and weakness. Other physiological biometrics – Hand scan – Retina scan – AFIS (Automatic Finger Print Identification Systems) – Behavioral Biometrics – Signature scan-keystroke scan.

MODULE 5 (7 hours)

Biometric Applications, Privacy, and Standards

Biometrics Application – Biometric Solution Matrix – Bio privacy – Comparison of privacy factor in different biometrics technologies – Designing privacy sympathetic biometric systems. Biometric standards – (BioAPI , BAPI) – Biometric middleware Biometrics for Network Security. Statistical

measures of Biometrics. Biometric Transactions.

Text Books

1. Biometrics – Identity Verification in a Networked World – Samir Nanavati, Michael Thieme, Raj Nanavati, WILEY- Dream Tech.
2. Biometrics for Network Security- Paul Reid, 1/e, Pearson Education.

Reference Books

1. Biometrics- The Ultimate Reference- John D. Woodward, Wiley
2. Biometrics for Network Security- Paul Reid, Pearson Education. Reference Books:
3. Introduction to Biometrics, Anil K. Jain, Arun A. Ross and Karthik Nanda kumar, 2011.
4. Guide to Biometrics, Ruud Bolle, Jonathan Connell, Sharanth Chandra Pankanti, Nalini Ratha and Andrew Senior, 2003.
5. Biometric Systems Technology, Design and Performance

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Foundations of Biometric Systems		7 Hours
1.1	Introduction to Biometrics	1
1.2	Benefits and Applications of Biometrics	1
1.3	Verification vs Identification	1
1.4	Biometric Matching Process	1
1.5	Accuracy Metrics: False Match Rate (FMR)	1
1.6	Accuracy Metrics: False Non-Match Rate (FNMR)	1
1.7	Failure to Enroll Rate and Derived Metrics	1
Module 2: Facial Recognition and Iris Recognition Technology		7 Hours
2.1	Facial Recognition: Features and Components	1
2.2	Facial Recognition Technologies and Limitations	1

2.3	Iris matching methods	1
2.4	Competing technologies, strengths and weaknesses	1
2.5	Introduction, iris features and uniqueness	1
2.6	Components of iris recognition system	1
2.7	Steps in iris recognition process	1
Module 3: Fingerprint Technology		7 Hours
3.1	Fingerprint Biometrics: Features and Components	1
3.2	Fingerprint Recognition Operation Steps	1
3.3	Fingerprint Recognition Technologies	1
3.4	Fingerprint Recognition Algorithms	2
3.5	Strengths and Weaknesses of Fingerprint Recognition	1
3.6	Types of algorithms used for interpretation.	1
Module 4: Voice Recognition and Multimodal Biometric Technologies		8 Hours
4.1	Voice Recognition: Features and Components	1
4.2	Voice Recognition Operation Steps	1
4.3	Voice Recognition Algorithms	1
4.4	Strengths and Weaknesses of Voice Biometrics	1
4.5	Hand Geometry Recognition Systems	1
4.6	Retina Recognition Systems	1
4.7	AFIS (Automatic Fingerprint Identification System)	1
4.8	Behavioral Biometrics: Signature Scan	1
Module 5: Biometric Applications, Privacy, and Standards		7 Hours
5.1	Behavioral Biometrics: Keystroke Dynamics	1
5.2	Applications of Biometrics in Security Systems	1
5.3	Biometric Solution Matrix	1
5.4	Technology Selection in Biometric Systems	1
5.5	Privacy Issues in Biometrics	1
5.6	Ethical and Legal Issues in Biometrics	1
5.7	Biometric Standards and Future Trends	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Define biometrics and explain its advantages over traditional authentication methods.
2. Differentiate between verification and identification in biometric systems.
3. Explain the biometric matching process with a suitable diagram.

Course Outcome 2 (CO 2):

1. Explain facial recognition technology and its operational steps.
2. Explain the working principle of iris recognition technology.
3. Describe the algorithms used in iris recognition systems.
4. Evaluate the effectiveness of iris recognition compared to other biometric technologies.

Course Outcome 3 (CO 3):

1. Compare fingerprint recognition and facial recognition technologies.
2. Discuss the strengths and limitations of fingerprint and facial biometric systems.
3. Explain the working principle of fingerprint recognition systems.
4. Describe the key features used in fingerprint recognition.

Course Outcome 4 (CO 4):

1. Discuss privacy concerns associated with biometric systems.
2. Explain ethical issues related to biometric data collection and usage.
3. Discuss legal and security challenges in biometric authentication systems.
4. Explain the role of biometric standards in ensuring secure biometric systems.

Course Outcome 5 (CO 5):

1. Explain the working of hand geometry and retina recognition systems.
2. Describe the Automated Fingerprint Identification System (AFIS).
3. Explain behavioral biometrics with examples such as signature scan and keystroke dynamics.
4. Discuss applications of biometrics in security systems.
5. Explain the biometric solution matrix and its role in technology selection.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

EIGHTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2028

Course Code: B24CS4P53

Course Name: BIOMETRICS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define biometrics and explain its advantages.
2. Differentiate between verification and identification.
3. What are the key components of a facial recognition system?
4. What are the main features used in iris recognition?
5. List the important features used in fingerprint recognition.
6. List the strength and weakness of fingerprint
7. Define voice biometrics.
8. What is AFIS?
9. What are biometric standards?
10. List applications of biometrics in security systems.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the biometric matching process and system architecture. (7)

- (b) A secure office building plans to implement biometric access control. Analyze the advantages of biometric authentication and recommend a suitable biometric system for this scenario. (7)

OR

12. (a) Discuss biometric accuracy metrics such as FMR, FNMR and Failure to Enroll. (7)
(b) A bank plans to introduce biometric authentication for ATM transactions. Suggest a suitable biometric modality and justify your choice considering accuracy, cost, and security. (7)
13. (a) A government plans to introduce an automated border control system using Iris Recognition to verify travelers at immigration checkpoints. Analyze how the iris recognition system can be designed for this application. Discuss the components of the system, steps in iris recognition, suitable feature extraction techniques, and matching methods used to ensure accurate identification. (7)
(b) Explain iris recognition architecture and operational steps. (7)

OR

14. (a) Discuss iris recognition algorithms and performance evaluation. A country plans to integrate iris recognition into its national identity program similar to Aadhaar. (7)
(b) Evaluate the advantages and challenges of using iris recognition for large-scale population identification. (7)
15. (a) Explain the working of fingerprint recognition technology. (7)
(b) A smartphone manufacturer wants to integrate fingerprint authentication. Design the working process of a fingerprint recognition system suitable for this application. (7)

OR

16. (a) Compare fingerprint recognition and facial recognition systems. (7)
(b) A smartphone manufacturer plans to integrate facial recognition for user authentication. Analyze the strengths and limitations of facial recognition in this application. (7)
17. (a) Explain voice recognition technology and its applications. (14)

OR

18. (a) Discuss physiological and behavioral biometrics with examples. (14)
19. (a) Discuss privacy and ethical issues in biometric technologies. (7)
(b) A government database plans to store biometric data of citizens. Analyze privacy concerns and propose secure solutions for biometric data protection. (7)

OR

20. (a) A national airport authority wants to implement a multi-modal biometric system. Propose a layered biometric security solution and justify the selected technologies. (7)
(b) Explain the biometric solution matrix and technology selection. (7)

B24CS4P54	PROGRAMMING IN R	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

The objective of this course is to enable the learner to make use of R Programming language to perform analysis and extraction of information from data irrespective of the quantity. It encompasses the R programming environment, syntax, data representations, data processing, statistical analysis and visualization. This course facilitates the learner to develop modular software solutions to perform statistical analysis and data extraction.

Prerequisites

Fundamental concepts in programming in C and Probability and Statistical Modeling

Course Outcomes

After the completion of the course the student will be able to:

CO1	Apply conditional and iterative constructs, functions, and data structures in R programming to solve computational problems. (Cognitive Knowledge Level: Apply)
CO2	Apply R programming techniques for data importing, cleaning, transformation, summarization, and preprocessing from various data sources. (Cognitive Knowledge Level: Apply)
CO3	Apply probability distributions, statistical techniques, and data analysis methods using R for solving engineering and real-world problems. (Cognitive Knowledge Level: Apply)
CO4	Analyze, visualize, and interpret data using graphical tools in R while considering ethical, societal, and professional aspects of data representation. (Cognitive Knowledge Level: Apply)
CO5	Develop and evaluate regression models using R for predictive analytics and data-driven decision making. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	2	2	2				1	1	
CO 2	2	2	2	3	2	1			2	2	2
CO 3	3	2	2	3	3				2	3	2
CO 4	2	2	3	3	3	3	3	3	3	2	3
CO 5	3	3	3	3	3	2		2	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (8 hours)

Introduction to R:

The R Environment - Command Line Interface and Batch processing, R Packages, Variables, Data Types, Vectors- vector operations and factor vectors, List- operations, Data Frames, Matrices and arrays, Control Statements- Branching and looping - For loops, While loops, Controlling loops. Functions- Function as arguments, Named arguments

MODULE 2 (8 hours)

Reading and writing data:

Importing data from Text files and other software, Exporting data, importing data from databases- Database Connection packages, Missing Data - NA, NULL Combining data sets, Transformations, Binning Data, Subsets, summarizing functions. Data Cleaning, Finding and removing Duplicates, Sorting.

MODULE 3 (6 hours)

Statistics with R

Analyzing Data, Summary statistics, Statistical Tests- Continuous Data, Discrete Data, Power tests, Common distributions- type arguments. Probability distributions, Normal distributions

MODULE 4 (6 hours)

Data Visualization

R Graphics- Overview, Customizing Charts, Graphical parameters, Basic Graphics functions, Lattice Graphics - Lattice functions, Customizing Lattice Graphics, Ggplot.

MODULE 5 (8 hours)

Regression Models

Building linear models - model fitting, Predict values using models, Analyzing the fit, Refining the model, Regression- types, Unusual observation and corrective measures, Comparison of models, Generalized linear models - Logistic Regression, Poisson Regression, Nonlinear least squares.

Text Books

1. Joseph Adler, " R in a Nutshell", Second edition, O'reilly, 2012

Reference Books

1. Jared P Lander, R for Everyone- Advanced analytics and graphics, Addison Wesley data analytics series, Pearson.
2. Norman matloff, The art of R programming, A Tour of Statistical, Software Design, O'reilly.
3. Robert Kabacoff, R in action, Data analysis and graphics with R, Manning.
4. Garret Grolemond, Hands-on programming with R, Write your own functions and simulations, O'reilly.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to R		8 hours
1.1	The R Environment – Command Line Interface, Batch processing, R Packages	1
1.2	Variables, Data Types	1
1.3	Vectors – Vector operations, Factor vectors	1
1.4	Lists – List operations, Data Frames	1
1.5	Matrices and Arrays	1
1.6	Control Statements – If, Else, Switch	1
1.7	Loops – For loops, While loops, Loop control	1
1.8	Functions – Function arguments, Named arguments	1
Module 2: Reading and Writing Data		8 hours
2.1	Importing data from text files and other software, Exporting data	1
2.2	Importing data from databases – Database connection packages	1
2.3	Missing Data – NA, NULL	1
2.4	Combining datasets, Data transformations	1
2.5	Binning data, Subsets, Summarizing functions	1
2.6	Data Cleaning	1
2.7	Finding and removing duplicates	1
2.8	Sorting	1

Module 3: Statistics with R		6 hours
3.1	Analyzing Data	1
3.2	Summary Statistics	1
3.3	Statistical Tests – Continuous data, Discrete data, Power tests	1
3.4	Common distributions – Type arguments	1
3.5	Probability distributions	1
3.6	Normal distribution	1
Module 4: Data Visualization		6 hours
4.1	R Graphics – Overview	1
4.2	Customizing charts	1
4.3	Graphical parameters, Basic graphics functions	1
4.4	Lattice Graphics – Lattice functions	1
4.5	Customizing lattice graphics	1
4.6	ggplot	1
Module 5: Regression Models		8 hours
5.1	Building linear models – Model fitting	1
5.2	Prediction using models, Model evaluation and refinement	2
5.3	Types of regression	1
5.4	Unusual observations and corrective measures	1
5.5	Comparison of models	1
5.6	Generalized linear models – Logistic and Poisson regression	1
5.7	Nonlinear least squares	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Write an R program to find the largest element in a vector using a loop.
2. Write an R program to print a multiplication table of a given number using for loops.

Course Outcome 2 (CO2):

1. Write an R program to merge two data frames based on a common column.
2. Write an R program to read a CSV file and display the first 5 rows.

Course Outcome 3 (CO3):

1. Write an R program to generate 20 random numbers from a normal distribution with mean = 50 and sd = 5.
2. Write an R program to calculate mean, median, and standard deviation of a numeric vector.

Course Outcome 4 (CO4):

1. Write an R program to create a bar plot for the following data: Category: A, B, C, D; Values: 10, 20, 15, 25.
2. Explain ggplot() and write an R program to plot a scatter plot of $x = 1:10$ vs $y = x^2$.

Course Outcome 5 (CO5):

1. Discuss the assumptions of linear regression and why they are important.
2. Describe the concept of nonlinear regression and give an example of a situation where it would be preferred over linear regression.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

EIGHTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2028

Course Code: B24CS4P54

Course Name: PROGRAMMING IN R

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Write an R program to generate a sequence of numbers from 1 to 100 with a step size of 5.
2. Explain vector in R with an example.
3. Explain why data cleaning is considered an important task in data analysis.
4. List different methods used in binning data.
5. What is the Wilcoxon test, and when is it used in R?
6. Explain the concept of normality testing in R.
7. Explain the t-test.
8. Explain how to plot a pie chart in R.
9. Explain box plots with the help of an R program.
10. Explain the process of fitting a linear regression model using the least squares method.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Write an R program to check whether a number is prime number or not. (4)

- (b) Explain the scope of variables in R functions with an example. (10)

OR

12. (a) Write an R program to compute the factorial of a number using iteration. (7)
(b) Write an R program to extract 3rd and 5th rows with 1st and 3rd columns from a data frame. (7)
13. (a) Illustrate transformation functions in R. (8)
(b) Write an R program to export the following data to a CSV file. (6)

id	name	gender	dob	country
10	Daisey	M	1990-10-02	Brazil
11	Harry	M	1981-03-24	Canada
12	Rachelle	F	1987-06-14	France
13	Zara	F	1985-08-16	Guinea

OR

14. (a) Write an R program to extract specific columns from a data frame and perform basic operations on them. (7)
(b) Explain how data is exported from a database in R programming. (7)
15. (a) Explain t-test and ANOVA in R. (7)
(b) Explain different statistical tests performed in continuous data. (7)

OR

16. (a) Explain the common distribution type arguments used in R functions for statistical analysis. (7)
(b) Explain different parametric tests in R. (7)
17. (a) Explain `ggplot()` with an example. (7)
(b) Write R program to create a pie chart for the following data. (7)

Category	Value
Housing	600
Food	300
Clothes	150
Entertainment	100
Others	200

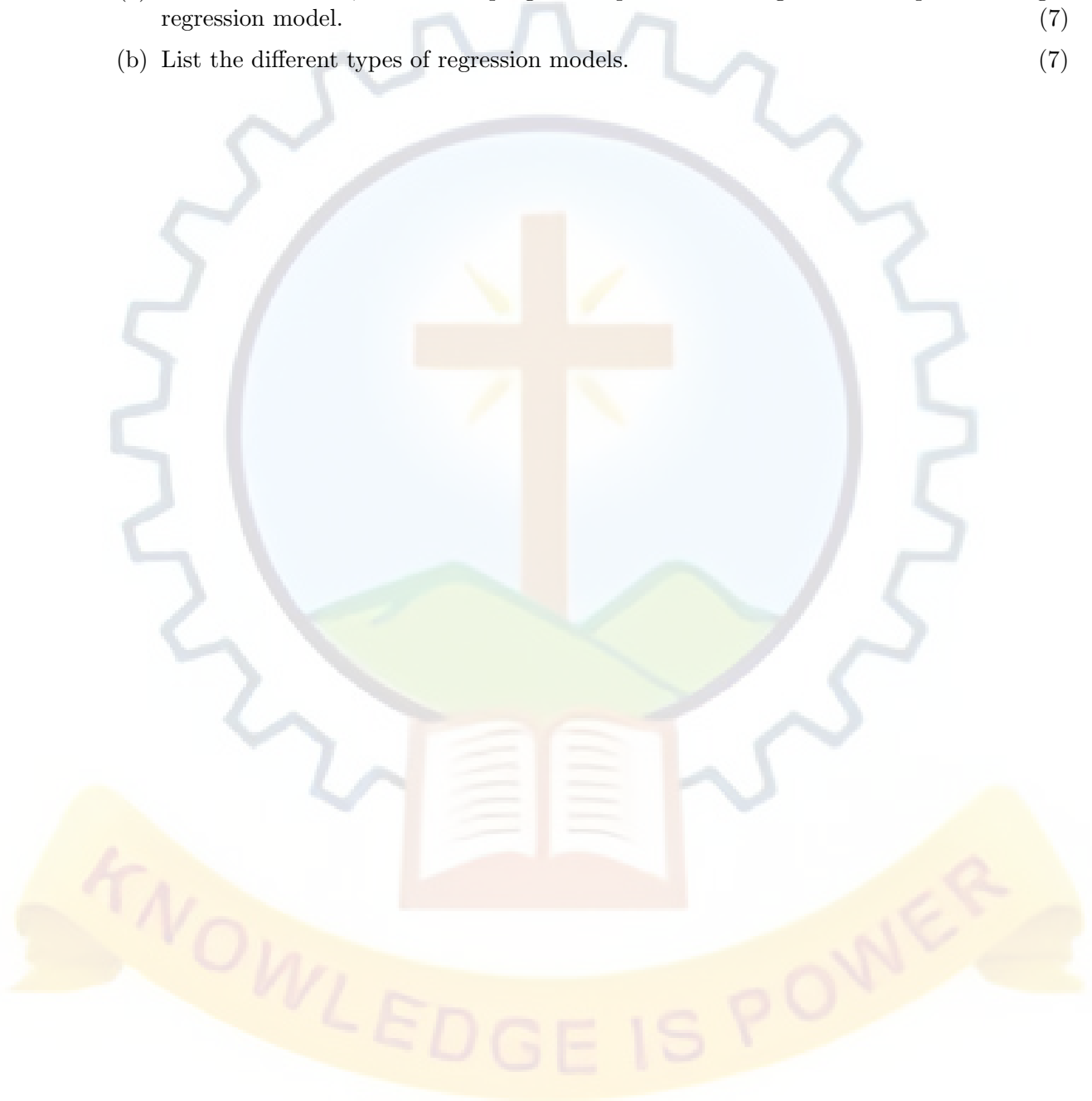
OR

18. (a) What is lattice graphics? Explain the working of lattice graphics. (7)
(b) Explain graphics devices used in data visualization.

19. (a) Distinguish between simple and multiple regression analysis and explain its applications when working with numerical and categorical data. (9)
(b) With the help of an example write the steps to establish a regression. (5)

OR

20. (a) Given two vectors, write an R program to predict the weight of a new person using a regression model. (7)
(b) List the different types of regression models. (7)



B24CS4P61	SOFTWARE TESTING PLATFORMS AND TOOLS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course aims to provide students with a strong foundation in software testing concepts, processes, and methodologies. It covers essential topics such as software quality, testing techniques, automation tools, security and performance testing, and modern trends including AI and GenAI in testing. It enables the learners to apply testing strategies, use standard tools, manage defects, and perform automated testing to build reliable and secure software.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO1	Understand the fundamental concepts of software testing, including importance, software quality, real-world failures, testing processes, levels of thinking, and basic testing terminologies. Also understand modern trends in software testing, including AI and Generative AI. (Cognitive Knowledge Level: Understand)
CO2	Apply security testing concepts, tools (OWASP, Burp Suite), graph-based testing techniques, and AI-driven methods for test case generation, optimization, and predictive testing to ensure application security and quality. (Cognitive Knowledge Level: Apply)
CO3	Understand test organization, planning, monitoring, defect management, and configuration management in software testing. (Cognitive Knowledge Level: Understand)
CO4	Understand the uses, types, benefits, risks, and effective implementation of software testing tools and test automation. (Cognitive Knowledge Level: Understand)
CO5	Apply testing techniques and automation tools and evaluate their impact on quality, security, and ethical considerations. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	3	2	2	2	3	3	3	2	2	3
CO 2	3	3	3	3	3	3	3	3	3	2	2
CO 3	2	3	3	3	2	3	3	3	3	3	2
CO 4	2	3	3	3	3	3	3	3	3	3	3
CO 5	3	3	3	3	3	3	3	3	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction to Software Testing and Automation

Introduction to Software Testing - Concepts, importance of testing, software quality, and real-world failures (e.g., Ariane 5, Therac 25); Software Testing Processes - Levels of thinking in testing; Testing Terminologies - Verification, validation, fault, error, bug, test cases and coverage criteria. Industry Trends - AI in test case automation, Introduction to GenAI in testing; Automation in Testing - Introduction to automation tools Case Study- Selenium,(Modern UI Testing) JUnit, Postman (API Testing) JMeter (Performance) OWASP ZAP (Security)

MODULE 2 (7 hours)

Security Testing and AI-Driven Automation:

Security Testing - Fundamentals, tools (OWASP, Burp Suite), and their role in protecting modern applications; Case Study - Application of graph based testing and security testing using industry standard tools. AI in Testing - GenAI for test case generation and optimization, impact on automation; Industry Tools - Application of AI-driven testing tools in automation and predictive testing; Case Study - Mutation testing using JUnit, AI-enhanced test case automation.

MODULE 3 (8 hours)

Test Management

Test Organization - Independent Testing,Roles, Tasks, and Qualifications. Testing Strategies-Test Planning, Selecting a Testing Strategy,Concrete Strategies,Testing and Risk,Testing Effort and Costs,Estimating Testing Effort,The Cost of Testing vs. The Cost of Defects. Test Planning , Control, and Monitoring - Test Execution Planning, Test Control,Test Cycle Monitoring,Test Reports. Defect Management- Evaluating Test Reports,Creating a Defect Report,Classifying Failures and Defects,Defect Status Tracking, Evaluation and Reporting. Configuration Management

MODULE 4 (6 hours)

Test Tools

Uses of test tools,Types of Test Tools-Test Management Tools,Test Specification Tools,Static Test Tools, Tools for Automating Dynamic Tests, Load and Performance Testing Tools,Tool-Based Support for Other Kinds of Tests. Benefits and Risks of Test Automation. Using Test Tools Effectively-Basic Considerations and Principles,Tool Selection, Pilot Project,Success Factors During Rollout and Use.

MODULE 5 (8 hours)

Advanced Testing Concepts and Test Automation and Tools

Performance Testing: Load Testing, Stress Testing, Volume Testing, Fail-over Testing, Recovery Testing, Compatibility Testing, Usability Testing, Testing the Documentation, Security Testing, Testing in the Agile Environment, Testing Web and Mobile Applications. Automated Software Testing, Automate Testing of web Applications, Selenium: Introducing Web Driver and Web Elements, Locating Web Elements, Actions on web elements, Different Web Drivers, Understanding Web Driver Events, Testing: Understanding Testing.xml, Adding classes, packages, Methods to Test, Test Reports

Text Books

1. Software-testing-foundations-a-study-guide-for-the-certified-tester-exam-5th-revised
2. Yogesh Singh, Software Testing, Cambridge University Press, 2012
3. Unmesh Gundecha, Satya Avasarala, Selenium WebDriver 3 Practical Guide – Second Edition, 2018
4. Paul Ammann and Jeff Offutt, Introduction to Software Testing, Cambridge University Press 2/e, 2016

Reference Books

1. Paul C. Jorgensen, Software Testing: A Craftsman's Approach, Fourth Edition, 2014, Taylor & Francis Group.
2. Software Testing Ron Patten Pearson 2/e, 2005
3. Carl Cocchiaro, Selenium Framework Design in Data-Driven Testing, 2018, Packt Publishing.
4. Elfriede Dustin, Thom Garrett, Bernie Gaurf, Implementing Automated Software Testing, 2009, Pearson Education, Inc.
5. Satya Avasarala, Selenium WebDriver Practical Guide, 2014, Packt Publishing.
6. Varun Menon, TestNg Beginner's Guide, 2013, Packt Publishing.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to Software Testing and Automation		7 hours
1.1	Introduction to Software Testing - Concepts, importance of testing, software quality	1

1.2	Real-world failures (e.g., Ariane 5, Therac 25); Software Testing Processes - Levels of thinking in testing	1
1.3	Testing Terminologies - Verification, validation, fault, error, bug, test cases, and coverage criteria;	2
1.4	Industry Trends - AI in test case automation, Introduction to GenAI in testing	1
1.5	Automation in Testing	1
1.6	Introduction to automation tools	1
1.7	Case Study- Selenium,(Modern UI Testing) JUnit, Postman (API Testing) JMeter (Performance) OWASP ZAP (Security)	1
Module 2: Unit Testing, Mutation Testing and AI-Driven Automation		7 hours
2.1	Security Testing - Fundamentals, tools (OWASP, Burp Suite)	1
2.2	Role in protecting modern applications	1
2.3	Case Study - Application of graph based testing and security testing using industry standard tools (e.g., Muclipse);	1
2.4	AI in Testing - GenAI for test case generation and optimization, impact on automation	1
2.5	Industry Tools - Application of AI-driven testing tools in automation and predictive testing;	1
2.6	Case Study - Mutation testing using JUnit, AI-enhanced test case automation	1
Module 3: Test Management		8 hours
3.1	Test Organization - Independent Testing,Roles, Tasks, and Qualifications.	1
3.2	Testing Strategies- Test Planning, Selecting a Testing Strategy,Concrete Strategies	1
3.3	Testing and Risk,Testing Effort and Costs,Estimating Testing Effort	1
3.4	The Cost of Testing vs. The Cost of Defects. Test Planning.	1
3.5	Control, and Monitoring - Test Execution Planning, Test Control,Test Cycle Monitoring,Test Reports.	1
3.6	Defect Management- Evaluating Test Reports,Creating a Defect Report	1
3.7	Classifying Failures and Defects,Defect Status Tracking, Evaluation and Reporting.	1
3.8	Configuration Management	1
Module 4: Test Tools		6 hours

4.1	Uses of test tools,Types of Test Tools-Test Management Tools	1
4.2	Test Specification Tools,Static Test Tools, Tools for Automating Dynamic Tests	1
4.3	Load and Performance Testing Tools,Tool-Based Support for Other Kinds of Tests.	1
4.4	Benefits and Risks of Test Automation.	1
4.5	Using Test Tools Effectively-Basic Considerations and Principles.	1
4.6	Tool Selection, Pilot Project,Success Factors During Rollout and Use.	1
Module 5: Advanced Testing Concepts and Test Automation and Tools		8 hours
5.1	Performance Testing: Load Testing, Stress Testing, Volume Testing, Fail-over Testing, Recovery Testing, Compatibility Testing, Usability Testing.	1
5.2	Testing the Documentation, Security Testing, Testing in the Agile Environment	1
5.3	Testing Web and Mobile Applications. Automated Software Testing.	1
5.4	Automate Testing of web Applications,Selenium	1
5.5	Introducing Web Driver and Web Elements, Locating Web Elements, Actions on web elements	1
5.6	Different Web Drivers, Understanding Web Driver Events.	1
5.7	Testing: Understanding Testing.xml, Adding classes, packages	1
5.8	Adding Methods to Test, Test Reports.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Describe the five levels of testing goals based on test process maturity.
2. Explain testing terminologies: test case, test suite, test coverage, and test data. Illustrate with examples.

Course Outcome 2 (CO 2):

1. An e-commerce application has high code coverage but still encounters logical failures in production. Design a mutation testing strategy to improve test effectiveness.

2. Discuss the application of AI-driven tools in predictive testing and automated test case generation.

Course Outcome 3 (CO 3):

1. Explain the role of configuration management in testing.
2. Explain how configuration management supports defect tracking and testing.

Course Outcome 4 (CO 4):

1. Evaluate different criteria for selecting a test tool in an organization.
2. Discuss the success factors during the rollout and usage of test tools in an organization.

Course Outcome 5 (CO 5):

1. Test the performance of the e-commerce application
2. Automate the testing of e-commerce application using Selenium

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

EIGHTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2028

Course Code: B24CS4P61

Course Name: SOFTWARE TESTING PLATFORMS AND TOOLS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define the terms failure, error, fault and defect in software testing.
2. List out any three popular software bugs .
3. What is mutation score, and how is it calculated.
4. Mention any three impacts of AI on test automation.
5. What is Symbol Round Trip Coverage in graph-based testing?
6. Define DU (Definition-Use) paths and DU pairs in data flow testing
7. What is functional testing, and what are the key steps involved in the process
8. Explain the difference between Equivalence Class Partitioning and Boundary Value Analysis?
9. Briefly explain three techniques of Grey box testing?
10. Explain the concept of symbolic execution with the help of a toy example?

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the importance of software testing. Illustrate with real-world failures such as Ariane 5 or Therac-25. (7)
(b) Differentiate between verification and validation. Also define fault, error, failure, and bug with examples. (7)

OR

12. (a) Define and differentiate: fault, error, failure, bug, and defect. (6)
(b) Explain the role of AI and Generative AI in modern software testing and test automation. (7)
13. (a) Explain graph-based testing techniques. Define node coverage, edge coverage, and path coverage with examples. (6)
(b) Describe mutation testing. How can AI be used to improve test case generation and optimization? (8)

OR

14. (a) Explain the working of Burp Suite in security testing. How does it help in detecting vulnerabilities in web applications? (7)
(b) Describe how AI can be used for test case generation, optimization, and predictive testing with suitable examples. (7)
15. (a) Explain test planning and test execution planning. What factors influence the selection of a testing strategy? (7)
(b) Describe the defect management process. How are defects classified and tracked? (7)

OR

16. (a) Describe defect management lifecycle with a neat diagram. (7)
(b) Explain configuration management and its importance in software testing (7)
17. (a) Explain different types of test tools such as test management tools, static tools, and dynamic testing tools. (7)
(b) Explain the process of tool selection in software testing. What are the criteria and challenges involved? (7)

OR

18. (a) Explain different categories of test tools:
i. Test specification tools
ii. Static testing tools
iii. Dynamic testing tools
with examples. (7)
(b) Discuss load and performance testing tools. Explain how tools support different kinds of testing. (7)
19. (a) Explain various types of performance testing such as load, stress, and volume testing with examples. (7)

- (b) Describe Selenium WebDriver. How are web elements located and interacted with in automated testing? (9)

OR

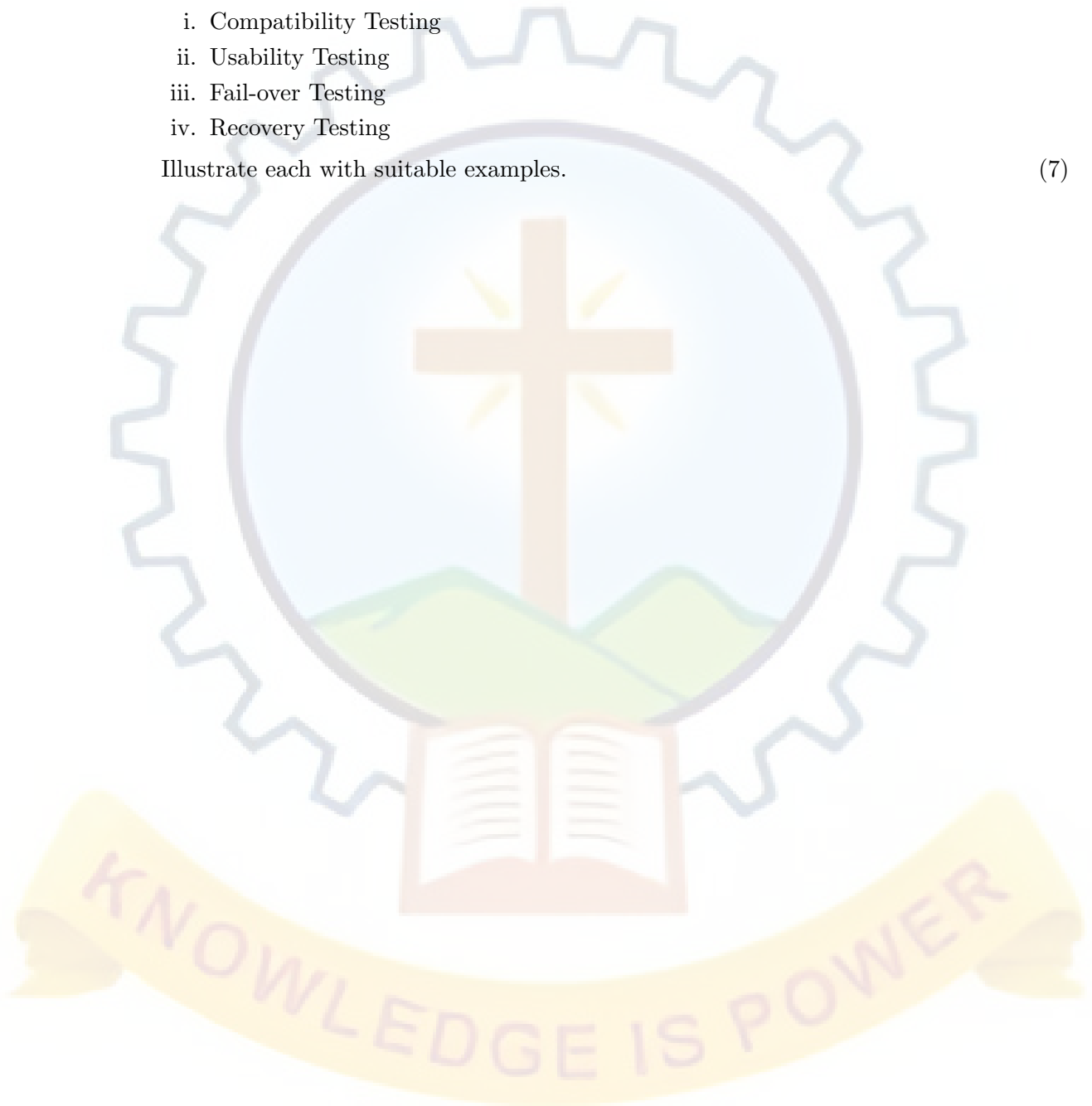
20. (a) Discuss testing in Agile environments. How does Agile testing differ from traditional testing approaches? Explain the role of continuous testing in Agile. (7)

- (b) Explain different types of testing in modern software systems:

- i. Compatibility Testing
- ii. Usability Testing
- iii. Fail-over Testing
- iv. Recovery Testing

Illustrate each with suitable examples.

(7)



B24CS4P62	BLOCKCHAIN TECHNOLOGIES	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course introduces the fundamental concepts of blockchain architecture, distributed ledgers and transaction mechanisms in decentralized networks. It covers consensus protocols used to validate transactions in trustless environments. The course examines cryptocurrencies such as Bitcoin and the principles underlying blockchain-based financial systems. It further presents Ethereum, smart contracts and decentralized applications.

Prerequisites

Basic knowledge of programming concepts, data structures and computer networks. **Course**

Outcomes

After the completion of the course the student will be able to:

Course Outcomes

After the completion of the course the student will be able to:

CO1	Explain the fundamental concepts, architecture, and components of blockchain technology and distributed ledger systems. (Cognitive Knowledge Level: Understand)
CO2	Illustrate the cryptographic building blocks and consensus mechanisms used in blockchain networks. (Cognitive Knowledge Level: Understand)
CO3	Explain the structure and operation of cryptocurrencies such as Bitcoin, including transactions and the mining process. (Cognitive Knowledge Level: Understand)
CO4	Demonstrate the working of the Ethereum platform, smart contracts, and decentralized applications. (Cognitive Knowledge Level: Apply)
CO5	Develop simple smart contracts using Solidity for blockchain-based applications. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2			1				1		1
CO 2	3	3	2	3	2	1	2		1		3
CO 3	2	3	2	3	2	3	3	1	2	2	3
CO 4	2	3	3	3	3	3	3	3	3	3	3
CO 5	3	3	3	3	3	3	3	3	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	20	20	20
Understand	50	50	50
Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study (Quiz/ seminar/ micro Project/ video assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 Hours)

Blockchain Fundamentals:

Introduction to Blockchain – Evolution of blockchain technology – Distributed ledger concept – Centralized vs decentralized systems – Structure of a blockchain – Blocks and block headers – Transactions and transaction verification – Hash functions and digital signatures – Peer-to-peer networks – Public, private and consortium blockchains – Advantages and limitations of blockchain technology.

MODULE 2 (7 Hours)

Cryptographic Foundations and Consensus Mechanisms:

Cryptographic primitives used in blockchain – Hash functions and properties – Merkle trees and their role in blockchain – Digital signatures – Byzantine Generals Problem – Consensus mechanisms in distributed systems – Proof of Work (PoW) – Proof of Stake (PoS) – Practical Byzantine Fault Tolerance (PBFT) – Comparison of consensus algorithms.

MODULE 3 (7 Hours)

Cryptocurrencies and Bitcoin:

Introduction to cryptocurrencies – Bitcoin architecture – Bitcoin network and nodes – Bitcoin transaction structure – Transaction lifecycle – Mining process – Mining difficulty and incentives – Hard fork and soft fork – Bitcoin wallets and key management – Security aspects of cryptocurrency systems.

MODULE 4 (7 Hours)

Ethereum and Smart Contracts:

Ethereum architecture – Ethereum accounts and transactions – Ethereum Virtual Machine (EVM) – Gas and transaction costs – Smart contracts – Features and lifecycle of smart contracts – Decentralized applications (DApps) – Development tools for Ethereum – Basic interaction with blockchain networks.

MODULE 5 (8 Hours)

Solidity Programming and Blockchain Applications:

Introduction to Solidity programming – Structure of a Solidity program – Data types, variables and functions – Modifiers and events – Control structures – Error handling – Deployment of smart contracts – Introduction to Hyperledger Fabric – Blockchain applications in financial systems,

healthcare data management, digital identity and supply chain tracking.

Text Books

1. Imran Bashir, Mastering Blockchain: A Deep Dive into Distributed Ledgers, Consensus Protocols, Smart Contracts, DApps, Cryptocurrencies, Ethereum and More, 3rd Edition, Packt Publishing, 2020.
2. Kumar Saurabh and Ashutosh Saxena, Blockchain Technology: Concepts and Applications, Wiley, 2020.
3. Chandramouli Subramanian, Asha A. George, Abhilash K. A., Meena Karthikeyan, Blockchain Technology, Universities Press, 2020.
4. Andreas M. Antonopoulos, Mastering Bitcoin: Programming the Open Blockchain, 2nd Edition, O'Reilly Media, 2017.
5. Chandramouli Subramanian, Asha A. George, Abhilash K. A., Meena Karthikeyan, Blockchain Technology, Universities Press, 2020. (Explains blockchain fundamentals, consensus mechanisms, and applications.)

Reference Books

1. Andreas M. Antonopoulos and Gavin Wood, Mastering Ethereum: Building Smart Contracts and DApps, O'Reilly Media, 2018.
2. Ritesh Modi, Solidity Programming Essentials: A Beginner's Guide to Build Smart Contracts for Ethereum and Blockchain, Packt Publishing, 2018.
3. Arshdeep Bahga and Vijay Madisetti, Blockchain Applications: A Hands-On Approach, VPT, 2017.
4. Don Tapscott and Alex Tapscott, Blockchain Revolution: How the Technology Behind Bitcoin Is Changing Money, Business, and the World, Penguin, 2016.
5. Melanie Swan, Blockchain: Blueprint for a New Economy, O'Reilly Media, 2015.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Blockchain Fundamentals		7 hours

1.1	Introduction to Blockchain and evolution of blockchain technology	1
1.2	Distributed ledger technology and decentralized systems	1
1.3	Blockchain architecture and components	1
1.4	Blocks, block headers and blockchain data structure	1
1.5	Blockchain transactions and transaction lifecycle	1
1.6	Peer-to-peer networking in blockchain	1
1.7	Types of blockchain networks and applications	1
Module 2: Blockchain Data Structures and Network Operations		7 hours
2.1	Blockchain data organization and block structure	1
2.2	Linking of blocks and formation of blockchain	1
2.3	Transaction structure and verification process	1
2.4	Block creation and block propagation in networks	1
2.5	Nodes and participants in blockchain networks	1
2.6	Forks in blockchain networks: hard fork and soft fork	1
2.7	Scalability and performance issues in blockchain	1
Module 3: Consensus Mechanisms in Blockchain		7 hours
3.1	Need for consensus in distributed systems	1
3.2	Distributed agreement problem and Byzantine Generals Problem	1
3.3	Byzantine Fault Tolerance concepts	1
3.4	Practical Byzantine Fault Tolerance (PBFT) algorithm	1
3.5	Phases of PBFT consensus protocol	1
3.6	Advantages and limitations of PBFT	1
3.7	Applications of PBFT in blockchain platforms	1
Module 4: Cryptocurrencies and Blockchain Platforms		7 hours
4.1	Cryptocurrency concepts and digital currencies	1
4.2	Bitcoin architecture and Bitcoin network	1
4.3	Bitcoin transactions and transaction lifecycle	1
4.4	Mining process and role of miners	1
4.5	Cryptocurrency wallets and key management	1
4.6	Ethereum architecture and Ethereum network	1
4.7	Ethereum Virtual Machine and decentralized applications	1

Module 5: Smart Contracts, Solidity and Blockchain Applications		8 hours
5.1	Smart contracts and decentralized applications	1
5.2	Blockchain oracles and external data integration	1
5.3	Ethereum platform and smart contract execution	1
5.4	Introduction to Solidity programming language	1
5.5	Solidity variables, data types and control structures	1
5.6	Functions, events and modifiers in Solidity	1
5.7	Deployment of smart contracts	1
5.8	Case studies: voting and auction smart contracts	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the concept of blockchain technology and discuss its evolution from centralized systems to decentralized distributed ledger systems.
2. Describe the structure of a blockchain. Illustrate the components of a block and explain how blocks are linked together to form a blockchain.

Course Outcome 2 (CO2):

1. Explain the blockchain transaction lifecycle. Discuss how transactions are validated and recorded in a blockchain network.
2. Describe the role of nodes in a blockchain network. Explain block creation, propagation and the occurrence of forks in blockchain systems.

Course Outcome 3 (CO3):

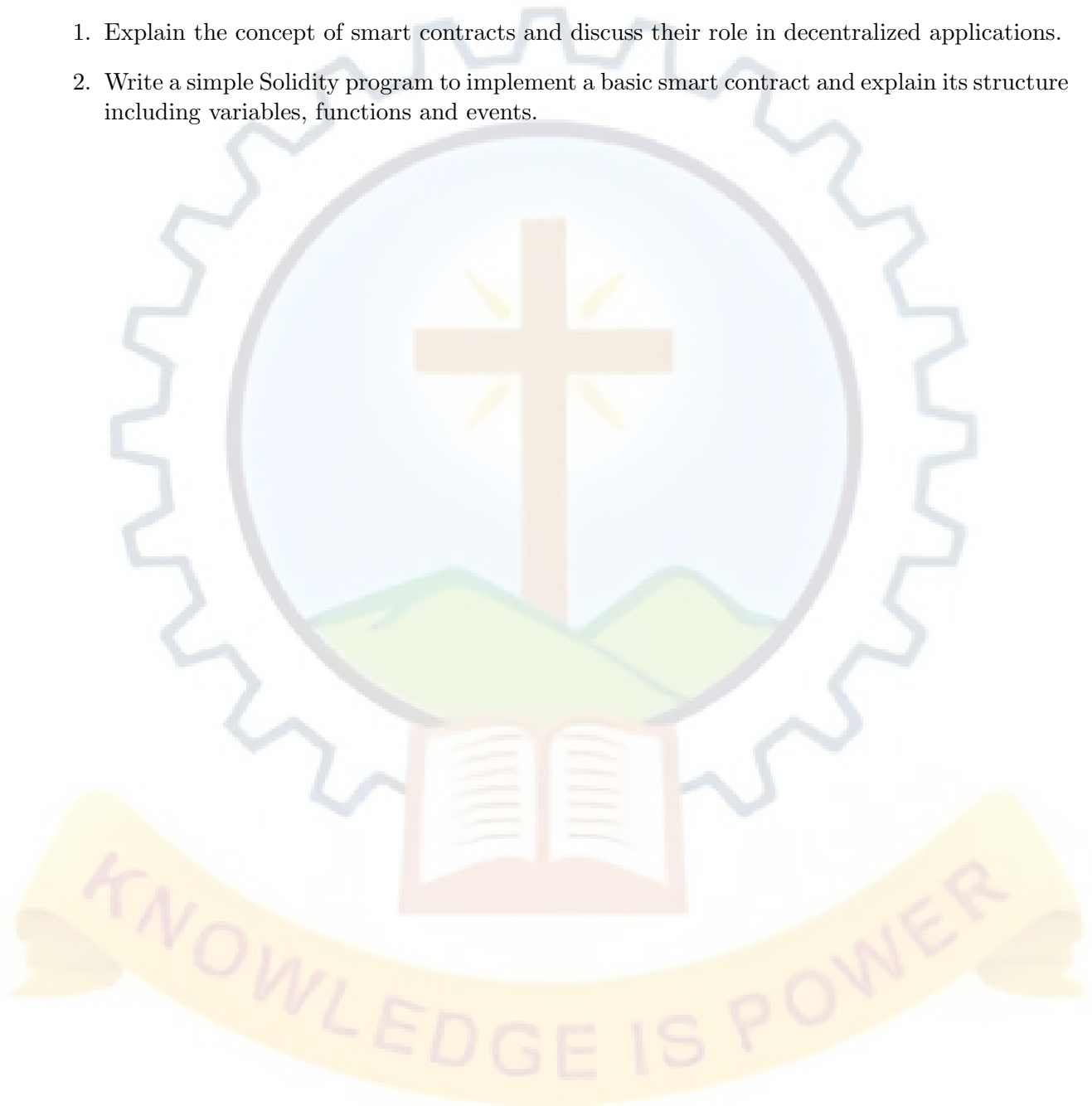
1. Explain the Byzantine Generals Problem and discuss the need for consensus in distributed blockchain networks.
2. Describe the Practical Byzantine Fault Tolerance (PBFT) consensus algorithm and explain its phases in achieving agreement among distributed nodes.

Course Outcome 4 (CO4):

1. Explain the architecture of Bitcoin and describe the lifecycle of a Bitcoin transaction.
2. Describe the architecture of the Ethereum platform. Explain the role of the Ethereum Virtual Machine (EVM) in executing decentralized applications.

Course Outcome 5 (CO5):

1. Explain the concept of smart contracts and discuss their role in decentralized applications.
2. Write a simple Solidity program to implement a basic smart contract and explain its structure including variables, functions and events.



MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

EIGHTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2028

Course Code: B24CS4P62

Course Name: BLOCKCHAIN TECHNOLOGIES

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Define blockchain technology and list any two characteristics of blockchain systems.
2. Differentiate between public and private blockchain networks.
3. What is a block header? List its main components.
4. Define hard fork and soft fork in blockchain networks.
5. What is the Byzantine Generals Problem?
6. List the phases involved in the Practical Byzantine Fault Tolerance (PBFT) algorithm.
7. What is cryptocurrency? Mention two characteristics of digital currencies.
8. Define Ethereum Virtual Machine (EVM).
9. What is a smart contract?
10. Define blockchain oracle and state its role.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the architecture of blockchain technology and describe the structure of a block. (7)
(b) Discuss the advantages and limitations of blockchain technology. (7)

OR

12. (a) Explain the working of blockchain networks with suitable diagrams. (7)
(b) Compare public, private and consortium blockchains. (7)
13. (a) Explain blockchain transaction lifecycle with suitable illustrations. (7)
(b) Describe block creation and propagation in blockchain networks. (7)

OR

14. (a) Explain the role of nodes in blockchain networks. (7)
(b) Discuss scalability challenges and forks in blockchain systems. (7)
15. (a) Explain the need for consensus in distributed blockchain systems. (7)
(b) Describe the Practical Byzantine Fault Tolerance (PBFT) consensus algorithm. (7)

OR

16. (a) Explain Byzantine Fault Tolerance and its significance in blockchain. (7)
(b) Discuss the advantages and limitations of PBFT consensus mechanism. (7)
17. (a) Explain the architecture of the Bitcoin network and the lifecycle of a Bitcoin transaction. (7)
(b) Describe the mining process and role of miners in cryptocurrency systems. (7)

OR

18. (a) Explain the architecture of Ethereum and its components. (7)
(b) Discuss the role of Ethereum Virtual Machine (EVM) in decentralized applications. (7)
19. (a) Explain the concept and lifecycle of smart contracts in blockchain systems. (7)
(b) Discuss the role of blockchain oracles in decentralized applications. (7)

OR

20. (a) Explain the structure of a Solidity program with suitable examples. (7)
(b) Describe the steps involved in deploying a smart contract in the Ethereum network. (7)

B24CS4P63	MOBILE COMPUTING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course introduces the fundamental concepts and technologies of mobile computing. It covers wireless communication architectures, protocols, and next-generation networks, enabling students to understand how mobile systems operate and communicate securely. The course also familiarizes learners with mobile applications, network layers, and the security aspects of modern wireless environments.

Prerequisites

A sound knowledge of computer networks.

Course Outcomes

After the completion of the course the student will be able to:

CO1	Explain the various mobile computing applications, services, design considerations, and architectures. (Cognitive Knowledge Level: Understand)
CO2	Describe the various technology trends for next-generation cellular wireless networks and apply the spreading concept in data transmission. (Cognitive Knowledge Level: Apply)
CO3	Summarize the architecture of various wireless LAN technologies. (Cognitive Knowledge Level: Understand)
CO4	Identify the functionalities of the mobile network layer, transport layer, and Wireless Application Protocol (WAP). (Cognitive Knowledge Level: Understand)
CO5	Interpret the security issues associated with mobile computing, next-generation technologies, and the fundamentals of 5G mobile networks. (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	2	1	1	2	1	1		1	1	2
CO 2	3	3	2	2	3			1	1	1	2
CO 3	2	2	2	2	3	1			1	1	2
CO 4	2	2	1	1	3	1			1	1	2
CO 5	2	2	1	1	2	2	3	1	1	2	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	50	50	50
Apply	20	20	20
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Mobile Computing Basics- Introduction to mobile computing – Functions, Middleware and Gateways, Application and services. Mobile computing architecture – Internet: The Ubiquitous network, Three- tier architecture for Mobile Computing, Design considerations for mobile computing.

MODULE 2 (10 hours)

Wireless Transmission and Communication Systems

Spread spectrum – Direct sequence, Frequency hopping. Medium Access Control – SDMA, FDMA, TDMA, CDMA. Satellite Systems – Basics, Applications, Geostationary Earth Orbit (GEO), Low Earth Orbit (LEO), Medium Earth Orbit (MEO), Routing, Localization, Handover. Telecommunication Systems - Global System for Mobile Communication (GSM) services, Architecture, Handover, Security.

MODULE 3 (6 hours)

Wireless LANs

Wireless LAN - Advantages, Design goals, Applications, Infra red vs radio, Infrastructure and Ad-hoc network, IEEE 802.11 - System Architecture, Protocol Architecture, Physical layer, Medium Access Control layer, Bluetooth - User scenarios, Architecture, Radio layer, Baseband layer, L2CAP, Profiles.

MODULE 4 (8 hours) Mobile Network and Transport Layer

Mobile network layer – Mobile Internet Protocol (IP), DHCP, Mobile ad-hoc networks. Mobile transport layer – Traditional Transmission Control Protocol (TCP), Improvements in Classical TCP. Wireless Application Protocol (WAP) - Architecture, Wireless Datagram Protocol (WDP), Wireless Transport Layer Security (WTLS), Wireless Transaction Protocol (WTP), Wireless Session Protocol (WSP).

MODULE 5 (6 hours) Mobile Security and Next Generation Networks

Security issues in mobile computing - Information security, Security techniques and algorithms, Security models. Next generation networks - Orthogonal Frequency Division Multiplexing (OFDM), Wireless Asynchronous Transfer Mode (WATM), Multi Protocol Label Switching (MPLS). 5G Mobile Networks - 10 pillars of 5G, Security for 5G communication.

Text Books

1. Asoke K. Talukder, Hasan Ahmad, Roopa R Yavagal, Mobile Computing Technology- Application and Service Creation, McGraw Hill Education, 2010.
2. Jochen Schiller, Mobile Communications, Pearson Education Asia, 2008.
3. Jonathan Rodriguez, Fundamentals of 5G Mobile Networks, Wiley Publishers, 2015.

Reference Books

1. Raj Kamal, Mobile Computing, 2/e, Oxford University Press.
2. Andrew S. Tanenbaum, Computer Networks, PHI, 3/e, 2003
3. Theodore S. Rappaport, Wireless Communications Principles and Practice, 2/e, PHI, New Delhi, 2004.
4. Curt M. White, Fundamentals of Networking and Communication 7/e, Cengage learning.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Mobile Computing Basics		6 hours
1.1	Introduction to mobile computing – Functions	1
1.2	Middleware and Gateways	1
1.3	Application and services	1
1.4	Mobile computing architecture – Internet: The Ubiquitous network	1
1.5	Three-tier architecture for Mobile Computing	1
1.6	Design considerations for mobile computing	1
Module 2: Wireless Transmission and Communication Systems		10 hours
2.1	Spread spectrum – Direct sequence	1
2.2	Spread spectrum – Frequency hopping	1
2.3	Medium Access Control – SDMA, FDMA	1
2.4	Medium Access Control – TDMA, CDMA	1
2.5	Satellite Systems – Basics, Applications	1

2.6	Geostationary Earth Orbit (GEO), Low Earth Orbit (LEO), Medium Earth Orbit (MEO)	1
2.7	Routing, Localization	1
2.8	Handover	1
2.9	Telecommunication Systems - GSM services, Architecture	1
2.10	GSM Handover, Security	1
Module 3: Wireless LANs		6 hours
3.1	Wireless LAN - Advantages, Design goals	1
3.2	Applications, Infra red vs radio	1
3.3	Infrastructure and Ad-hoc network	1
3.4	IEEE 802.11 - System Architecture, Protocol Architecture	1
3.5	Physical layer, Medium Access Control layer	1
3.6	Bluetooth - User scenarios, Architecture, Radio layer, Baseband layer, L2CAP, Profiles	1
Module 4: Mobile Network and Transport Layer		8 hours
4.1	Mobile network layer – Mobile Internet Protocol (IP)	1
4.2	DHCP	1
4.3	Mobile ad-hoc networks	1
4.4	Traditional Transmission Control Protocol (TCP)	1
4.5	Improvements in Classical TCP	1
4.6	WAP - Architecture, Wireless Datagram Protocol (WDP)	1
4.7	Wireless Transport Layer Security (WTLS), Wireless Transaction Protocol (WTP)	1
4.8	Wireless Session Protocol (WSP)	1
Module 5: Mobile Security and Next Generation Networks		6 hours
5.1	Security issues in mobile computing - Information security	1
5.2	Security techniques and algorithms, Security models	2
5.3	Next generation networks - OFDM, WATM	1
5.4	MPLS	1
5.5	5G Mobile Networks - 10 pillars of 5G, Security for 5G communication	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1)

1. Explain mobile computing architecture in detail.
2. Define mobile computing and explain its core components.

Course Outcome 2 (CO 2)

1. Explain spread spectrum techniques used in wireless communication.
2. Describe satellite communication systems used in mobile computing.

Course Outcome 3 (CO 3)

1. Explain the IEEE 802.11 wireless LAN architecture.
2. Compare infrastructure mode and ad-hoc mode in wireless LANs.

Course Outcome 4 (CO 4)

1. Explain Mobile IP and DHCP in mobile networks.
2. Describe mobile ad-hoc network routing protocols.

Course Outcome 5 (CO 5)

1. Discuss security issues in mobile computing.
2. Explain next-generation network technologies.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

EIGHTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2028

Course Code: B24CS4P63

Course Name: MOBILE COMPUTING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Why is middleware considered a critical component in mobile computing environments?
2. How does the three-tier architecture help in handling scalability issues in mobile applications?
3. Why are spread spectrum techniques preferred in hostile or noisy wireless environments?
4. What challenges do LEO satellite systems face compared to GEO satellites in mobile communication?
5. Why is ad-hoc mode more suitable for disaster recovery scenarios than infrastructure mode?
6. How does the MAC layer in IEEE 802.11 reduce collision in wireless communication?
7. Why does traditional TCP perform poorly in mobile wireless networks?
8. How does Dynamic Source Routing reduce routing overhead in mobile ad-hoc networks?
9. Why is key management more complex in mobile computing compared to wired networks?
10. How does OFDM improve spectral efficiency in next-generation wireless networks?

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Analyze how such architectures help manage mobility, scalability, and service availability in mobile computing systems. (7)
- (b) Examine how these constraints influence the design of mobile applications and supporting infrastructure. (7)

OR

12. (a) Discuss how middleware and gateways help overcome this challenge. (7)
 - (b) Justify this statement with reference to mobile computing architecture and services. (7)
13. (a) Discuss how spread spectrum and multiple access techniques address these challenges. (7)
 - (b) Analyze the trade-offs involved in choosing between GEO, LEO, and MEO satellite systems. (7)

OR

14. (a) Explain how handover and localization are handled in wireless and satellite communication systems. (7)
 - (b) Discuss this statement with reference to GSM architecture and multiple access techniques (7)
15. (a) Evaluate the reasons for this shift and the limitations that still exist. (7)
 - (b) Analyze infrastructure mode and ad-hoc mode with suitable use cases.. (7)

OR

16. (a) Discuss how the IEEE 802.11 MAC layer addresses collision and fairness issues. (7)
 - (b) Justify this statement with reference to Bluetooth. (7)
17. (a) Discuss the modifications and additional mechanisms required to support mobile users. (7)
 - (b) Analyze this statement with reference to DSR and DSDV protocols. (7)

OR

18. (a) Examine how this affects TCP performance in mobile environments and how improvements address the issue. (7)
 - (b) Discuss how WAP architecture achieves this adaptation. (7)
19. (a) Analyze the security challenges unique to mobile computing environments. (7)
 - (b) Discuss the reasons and explain suitable security models for mobile computing. (7)

OR

20. (a) Explain the technologies such as OFDM and MPLS. (7)
- (b) Discuss the pillars of 5G and analyze the associated security requirements. (7)

B24CS4P64	SOFT COMPUTING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

The purpose of this course is to introduce students to the fundamental concepts and techniques of soft computing. The course content covers fuzzy logic, evolutionary algorithms, swarm intelligence, and hybrid intelligent systems for modeling and analysis. By studying this course, students will develop the ability to solve complex engineering problems involving uncertainty and optimization.

Prerequisite: NIL

Course Outcomes

After the completion of the course the student will be able to:

CO1	Explain the fundamental concepts of soft computing and fuzzy sets for representing uncertainty and reasoning. (Cognitive Knowledge Level: Understand)
CO2	Analyze fuzzy logic concepts and fuzzy inference mechanisms for modeling, reasoning, and control. (Cognitive Knowledge Level: Apply)
CO3	Apply Genetic Algorithm techniques and multi-objective optimization techniques for solving optimization problems. (Cognitive Knowledge Level: Apply)
CO4	Apply nature-inspired swarm intelligence techniques for solving optimization problems. (Cognitive Knowledge Level: Apply)
CO5	Develop and implement solutions using fuzzy logic and nature-inspired optimization techniques, incorporating efficient use of computational resources and exploring relevant methods from current literature. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	1									
CO 2	2	2	3	3	3						
CO 3	2	3	3	3	3				1		2
CO 4	3	3	3	3	3				1		2
CO 5	3	3	3	3	3	3	3	3	2	2	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study (Seminar/Quiz/ micro Project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (6 hours)

Introduction to Soft Computing

Introduction to Soft Computing, difference between hard and soft computing, applications of soft computing. Crisp sets, fuzzy sets. Properties of fuzzy sets and operations on fuzzy sets. Fuzzy relations, Fuzzy membership functions.

MODULE 2 (8 hours)

Fuzzy Logic

Methods of membership value assignment methods. Fuzzy propositions and fuzzy implications. Fuzzy interpretations-Mamdani, Zadeh, Defuzzification methods- Centroid method, Centre of Sums, Mean of Maxima. Fuzzy inference System - mamdani and sugeno.

MODULE 3 (8 hours)

Genetic Algorithms

Introduction to Genetic Algorithms and their theoretical foundation. Gene representation and fitness function. Genetic operators: selection, crossover (recombination), mutation, and elitism. Applications of Genetic Algorithms. Multi-objective Genetic Algorithms: Pareto-optimal ranking, Goldberg's ranking, Fonseca-Fleming method, and Lu & Yen method.

MODULE 4 (7 hours)

Ant Colony Optimization

Nature-Inspired Computing. Introduction to Ant Colony Optimization (ACO). Development of ACO algorithms: pheromone updating, pheromone evaporation, and pheromone intensification. Types of Ant Colony models and applications.

MODULE 5 (7 hours)

Swarm Intelligence Techniques

Particle Swarm Optimization (PSO): concept, velocity and position update mechanisms, applications. Bee Colony Optimization (BCO): foraging behavior, algorithm structure, and applications. Comparative analysis of ACO, PSO, and BCO. Real-world applications of swarm intelligence techniques.

Text Books

1. Samir Roy, Udit Chakraborty, Introduction to Soft Computing Neuro-Fuzzy Genetic Algorithms, Pearson, 2013.
2. N.P. Padhy, Artificial Intelligence and Intelligent Systems, Oxford Press, New Delhi, 2005.
3. Dilip K. Pratihari, Soft Computing and AI: Fundamentals and Applications , Narosa Publishing House, 2025
4. Andries P. Engelbrecht, Computational Intelligence: An Introduction, 2007
5. S.N.Sivanandam and S.N. Deepa, Principles of Soft Computing , 2nd Edition, John Wiley & Sons, 2011.

Reference Books

1. Kalyanmoy Deb, Multi-objective Optimization using Evolutionary Algorithms, 1st Edition, John Wiley & Sons, 2009.
2. Timothy J Ross, Fuzzy Logic with Engineering Applications, John Wiley & Sons, 2016.
3. T.S.Rajasekaran, G.A.Vijayalakshmi Pai “Neural Networks, Fuzzy Logic & Genetic Algorithms Synthesis and Applications”, Prentice-Hall India.
4. Simon Haykin, “Neural Networks- A Comprehensive Foundation”, 2/e, Pearson Education.
5. Zimmermann H. J, “Fuzzy Set Theory & Its Applications”, Allied Publishers Ltd.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to Soft Computing		6 hours
1.1	Introduction to Soft Computing	1
1.2	Difference between Hard and Soft Computing	1
1.3	Applications of Soft Computing	1
1.4	Crisp Sets and Fuzzy Sets	1
1.5	Properties and operations of Fuzzy Sets	1
1.6	Fuzzy relation and membership function	1
Module 2: Fuzzy Logic		8 hours
2.1	Fuzzy membership value assignment - introduction	1

2.2	Membership Value Assignment	1
2.3	Fuzzy proposition	1
2.4	Fuzzy Implications	1
2.5	Fuzzy Interpretations – Mamdani and Zadeh	1
2.6	Defuzzification – Centroid Method	1
2.7	Centre of Sums Method and mean of maxima	1
2.8	Fuzzy inference system - mamdani and sugeno	1
Module 3: Genetic Algorithms		8 hours
3.1	Introduction and Theoretical Foundation of GA	1
3.2	Gene Representation and Fitness Function	1
3.3	Selection Techniques in GA	1
3.4	Crossover (Recombination) Methods	1
3.5	Mutation and Elitism	1
3.6	Applications of Genetic Algorithms	1
3.7	Multi-objective GA – Pareto and Goldberg Ranking	1
3.8	Fonseca–Fleming and Lu & Yen Methods	1
Module 4: Ant Colony Optimization		7 hours
4.1	Nature-Inspired Computing	1
4.2	Introduction to Ant Colony Optimization	1
4.3	Pheromone Updating Mechanism	1
4.4	Pheromone Evaporation	1
4.5	Pheromone Intensification	1
4.6	Types of Ant Colony Models	1
4.7	Applications of ACO	1
Module 5: Swarm Intelligence Techniques		7 hours
5.1	Particle Swarm Optimization – Concept	1
5.2	Velocity and Position Update Mechanisms	1
5.3	Applications of PSO	1
5.4	Bee Colony Optimization – Foraging Behavior	1
5.5	Algorithm Structure of BCO	1
5.6	Comparative Analysis – ACO, PSO, BCO	1
5.7	Real-world Applications of Swarm Intelligence	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Explain the concept of Soft Computing. Compare Hard Computing and Soft Computing with suitable examples.
2. Define crisp sets and fuzzy sets. Explain the properties and basic operations on fuzzy sets.

Course Outcome 2 (CO2):

1. Explain different types of fuzzy membership functions. Describe the methods used for assigning membership values such as intuition, inference, and rank ordering.
2. There is an imprecise relationship between the ambient temperature for clay masonry bricks and their compressive strengths. Let X be a fuzzy set of fracture strengths and Y be a fuzzy set of temperatures with the following membership functions:

$$X = \left\{ \frac{1.0}{1500} + \frac{0.8}{2175} + \frac{0.6}{7000} + \frac{0.5}{12750} + \frac{0.3}{16500} + \frac{0.1}{20000} \right\}$$

$$Y = \left\{ \frac{0.2}{20} + \frac{0.4}{25} + \frac{0.5}{32} + \frac{1.0}{50} + \frac{0.6}{90} + \frac{0.3}{105} \right\}$$

(a) Find the Cartesian Product of X and Y and represent it as relation R .

Suppose there is a second fuzzy set of masonry lengths given as:

$$Z = \left\{ \frac{0.4}{1500} + \frac{0.5}{2175} + \frac{0.6}{7000} + \frac{0.8}{12750} + \frac{0.9}{16500} + \frac{1.0}{20000} \right\}$$

- (b) Find $S = Z \circ R$ using max-min composition.
- (c) Find $T = Z \circ R$ using max-product composition.

Course Outcome 3 (CO3):

1. Explain the working principle of a Genetic Algorithm. Discuss the roles of selection, crossover, mutation, and elitism in the search process.
2. Using Genetic Algorithm with Roulette wheel selection method maximize the function $f(x) = x^2$ over $\{0, 1, 2, \dots, 31\}$ with initial x values (13, 24, 8, 19). Show one crossover and mutation step.

Course Outcome 4 (CO4):

1. Explain the basic principles of Ant Colony Optimization. Discuss pheromone updating, pheromone evaporation, and pheromone intensification.
2. Describe the different types of Ant Colony Optimization models and explain how swarm intelligence helps in solving optimization problems.

Course Outcome 5 (CO5):

1. Explain the concept of Particle Swarm Optimization (PSO) and describe how velocity and position of particles are updated.
2. Derive and explain the velocity update equation in PSO, highlighting the roles of inertia, cognitive, and social components.



MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

EIGHTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2028

Course Code: B24CS4P64

Course Name: SOFT COMPUTING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

- 1 Define Soft Computing. List any two differences between Hard Computing and Soft Computing.
- 2 What are fuzzy sets? Explain any two properties or operations on fuzzy sets.
- 3 What are fuzzy membership functions? Briefly explain any two methods of membership value assignment.
- 4 Define fuzzy relations. What are the basic operations that can be performed on fuzzy relations?
- 5 What is a Genetic Algorithm? List the main steps involved in a GA.
- 6 What are selection and mutation operators in Genetic Algorithms? Briefly explain their roles.
- 7 What is Ant Colony Optimization (ACO)? Explain the concept of pheromone update.
- 8 What is pheromone evaporation in ACO? Why is it important in the search process?
- 9 Describe the working of Bee Colony Optimization (BCO) with reference to foraging behavior and division of labor among bees.
- 10 Compare PSO and BCO algorithms in terms of search strategy, convergence speed, and applications.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Given two fuzzy sets $A = \{0.3/x_1 + 0.6/x_2 + 0.8/x_3 + 0.4/x_4\}$ and $B = \{0.5/x_1 + 0.2/x_2 + 0.7/x_3 + 0.9/x_4\}$, apply fuzzy set operations to find: (i) Union (ii) Intersection (iii) Complement of A. (8)
- (b) Apply the concepts of Hard Computing and Soft Computing to analyze how uncertainty in real-world problems (such as weather prediction or medical diagnosis) can be handled using soft computing techniques. (6)

OR

12. (a) Construct membership functions for the linguistic variable “Water Level” with labels Low, Medium, and High using triangular membership functions. Explain how membership values are assigned. (8)
- (b) Demonstrate the properties of fuzzy sets (commutativity, associativity, and distributivity) by applying them to suitable fuzzy sets. (6)
13. (a) Given fuzzy sets $A = \{0.4/x_1 + 0.7/x_2 + 1/x_3\}$ and $B = \{0.6/y_1 + 0.5/y_2 + 0.9/y_3\}$, construct the fuzzy relation $R = A \times B$ and represent it as a matrix. (8)
- (b) Apply the concept of λ -cuts to the fuzzy set $A = \{0.2/x_1 + 0.6/x_2 + 0.8/x_3 + 0.4/x_4\}$ for $\lambda = 0.5$ and interpret the result. (6)

OR

14. (a) Given two fuzzy relations $R(X, Y)$ and $S(Y, Z)$ represented as matrices, apply max–min composition to obtain the relation $T(X, Z)$. (8)
- (b) Demonstrate how defuzzification converts fuzzy output into a crisp value using any suitable defuzzification method. (6)
15. (a) Apply the steps of a Genetic Algorithm to maximize the function $f(x) = x^2$ over the range $\{0, 1, 2, \dots, 31\}$ starting with the population (5, 7, 10, 12). Show one iteration of selection and crossover. (10)
- (b) Demonstrate how mutation and elitism help maintain diversity and improve solution quality in Genetic Algorithms. (4)

OR

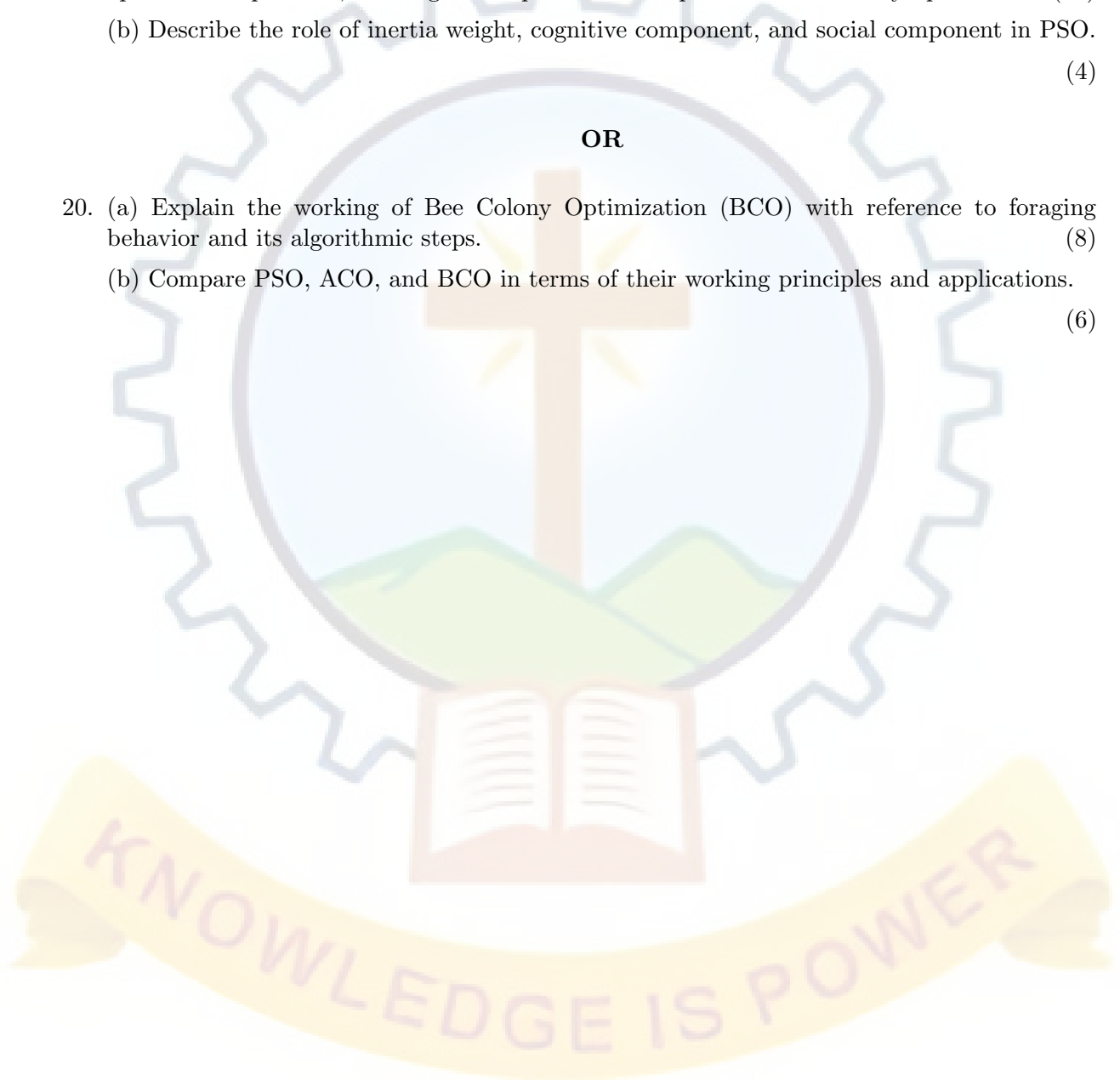
16. (a) Apply single-point crossover to the following chromosomes: Parent 1: 110101 Parent 2: 101011 Show the offspring generated. (8)
- (b) Explain how Pareto optimal ranking is applied in multi-objective Genetic Algorithms. (6)
17. (a) Apply the Ant Colony Optimization algorithm to determine the shortest path between nodes in a simple graph. Explain how ants update pheromone trails during the search process. (10)
- (b) Demonstrate the role of pheromone evaporation and pheromone intensification in guiding ants toward optimal solutions. (4)

OR

18. (a) Explain how swarm intelligence principles are applied in Ant Colony Optimization to solve combinatorial optimization problems such as the Traveling Salesman Problem. (8)
(b) Describe different types of Ant Colony models and explain how pheromone updating improves solution quality. (6)
19. (a) Explain the working of Particle Swarm Optimization (PSO) and apply it to solve a simple optimization problem, showing the steps involved in position and velocity updates. (10)
(b) Describe the role of inertia weight, cognitive component, and social component in PSO. (4)

OR

20. (a) Explain the working of Bee Colony Optimization (BCO) with reference to foraging behavior and its algorithmic steps. (8)
(b) Compare PSO, ACO, and BCO in terms of their working principles and applications. (6)



B24CS4G31	INTRODUCTION TO OPERATING SYSTEMS	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4	3	2024

Preamble

This course aims to understand the overall working of computer system, tradeoffs between performance and functionality and the division of jobs between hardware and software. This course introduces the concepts of memory management, device management, process management, file management and security and protection mechanisms available in an operating system. The course helps the learner to understand the fundamentals about any operating system design so that they can extend the features of operating system to detect and solve many problems occurring in operating system and to manage the computer resources appropriately.

Prerequisites

NIL

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Explain the relevance, structure and functions of Operating Systems in computing devices, concepts of process management and process scheduling mechanisms employed in Operating Systems. (Cognitive knowledge: Understand)
CO 2	Apply process synchronization concepts in an operating system by implementing and demonstrating synchronization mechanisms such as Mutex Locks, Semaphores, and Monitors to control concurrent process execution. (Cognitive knowledge: Apply)
CO 3	Apply appropriate deadlock handling techniques by implementing one method each for deadlock detection, prevention, avoidance, and recovery in an operating system scenario. (Cognitive knowledge: Apply)
CO 4	Apply appropriate memory management algorithms in an operating system to allocate, manage, and optimize main memory for executing processes. (Cognitive knowledge: Apply)
CO 5	Apply file and storage security concepts by implementing suitable security mechanisms and algorithms to protect data and manage secure access in an operating system environment. (Cognitive knowledge: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	2	1						1		1
CO 2	2	3	3	1	1						
CO 3	3	3	3	1	1						
CO 4	3	3	3	1	1						
CO 5	2	3	3	1	1	1					

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(Quiz/Seminar/Micro project/ Video assignments/ Analytical Assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction

Operating system overview – Operations, Functions, Service – System calls, Types – Operating System structure - Simple structure, Layered approach, Microkernel, Modules – System boot process.

MODULE 2 (9 hours)

Processes

Process states, Process control block, threads, scheduling, Operations on processes - process creation and termination – Inter-process communication - shared memory systems, Message-passing systems.

Process Scheduling

Basic concepts- Scheduling criteria -scheduling algorithms- First come-First Served, Shortest Job First, Priority scheduling, Round robin scheduling.

MODULE 3 (8 hours)

Process synchronization

Race conditions – Critical section problem – Peterson's solution, Synchronization hardware, Mutex Locks, Semaphores, Monitors – Synchronization problems - Producer Consumer, Dining Philosophers and Readers-Writers.

Deadlocks

Necessary conditions, Resource allocation graphs, Deadlock prevention, Deadlock avoidance – Banker's algorithms, Deadlock detection, Recovery from deadlock.

MODULE 4 (7 hours)

Memory Management

Concept of address spaces, Swapping, Contiguous memory allocation, fixed and variable partitions, Segmentation, Paging. Virtual memory, Demand paging, Page replacement algorithms.

MODULE 5 (5 hours)

File System:

File concept - Attributes, Operations, types, structure – Access methods, Protection. File-system implementation, Directory implementation. Allocation methods.

Storage Management:

Magnetic disks, Solid-state disks, Disk Structure, Disk scheduling, Disk formatting.

Text Books

Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, ' Operating System Concepts' 10th Edition, Wiley India 2018.

Reference Books

1. Andrew S Tanenbaum, "Modern Operating Systems" , 4th Edition, Prentice Hall, 2015.
2. William Stallings, "Operating systems", 6th Edition, Pearson, Global Edition, 2015.
3. Garry Nutt, Nabendu Chaki, Sarmistha Neogy, "Operating Systems", 3rd Edition, Pearson Education.
4. D.M.Dhamdhare, "Operating Systems", 2nd Edition, Tata McGraw Hill, 2011.
5. Sibsankar Halder, Alex A Aravind, "Operating Systems", Pearson Education.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction		5 hours
1.1	Introduction to Operating System	1
1.2	Operating System operations, functions, service	1
1.3	System calls, Types	1
1.4	Operating System Structure: Simple, Layered, Microkernel, Modules	1
1.5	System Boot Process	1

Module 2: Processes and Process Scheduling		7 hours
2.1	Processes, Process states, Process Control Block, Threads	1
2.2	Scheduling	1
2.3	Operations on processes: process creation and termination	1
2.4	Inter-process communication: Shared memory systems, Message Passing	1
2.5	Process Scheduling – Basic concepts, Scheduling Criteria, Scheduling algorithms - Basics	1
2.6	First come First Served, Shortest Job First	1
2.7	Priority scheduling, Round Robin Scheduling	1
Module 3: Process synchronization and Dead locks		9 hours
3.1	Process synchronization, Race conditions	1
3.2	Critical Section problem, Peterson's solution	1
3.3	Synchronization hardware, Mutex Locks	1
3.4	Semaphores, Monitors	1
3.5	Synchronization problem examples	1
3.6	Deadlocks: Necessary conditions, Resource Allocation Graphs	1
3.7	Deadlock prevention, Deadlock avoidance	1
3.8	Banker's algorithm	1
3.9	Deadlock detection, Deadlock recovery	1
Module 4: Memory Management		8 hours
4.1	Memory Management: Concept of Address spaces	1
4.2	Swapping	1
4.3	Contiguous memory allocation, fixed and variable partitions	1
4.4	Segmentation	1
4.5	Paging (Lecture 1)	1
4.6	Virtual memory, Demand Paging	1
4.7	Page replacement algorithms (Lecture 1)	1
4.8	Page replacement algorithms (Lecture 2)	1
Module 5: File and Disk management		7 hours
5.1	File concept, Attributes, Operations, types, structure	1
5.2	Access methods	1
5.3	Protection	1
5.4	File-System implementation, Directory implementation	1

5.5	Allocation methods	1
5.6	Magnetic disks, Solid-state disks, Disk structure	1
5.7	Disk scheduling, Disk formatting	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. What is the main advantage of the microkernel approach to system design? How do user programs and system programs interact in a microkernel architecture?
2. Define a process. With the help of a neat diagram, explain the different states of a process.

Course Outcome 2 (CO2):

1. What do you mean by binary semaphore and counting semaphore? With C code, explain the implementation of `wait()` and `signal()` operations.

Course Outcome 3 (CO3):

1. Describe a Resource Allocation Graph (RAG) for the following cases:
 - (a) With a deadlock.
 - (b) With a cycle but no deadlock.

Course Outcome 4 (CO4):

1. Consider the page reference string:
1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6
Find the number of page faults if there are four page frames using the following page replacement algorithms:
 - (a) LRU
 - (b) FIFO
 - (c) Optimal

Course Outcome 5 (CO5):

1. Explain the different file allocation methods with their advantages and disadvantages.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

EIGHTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2028

Course Code: B24CS4G31

Course Name: INTRODUCTION TO OPERATING SYSTEMS

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. How does hardware find the Operating System kernel after system switch-on?
2. What is the purpose of system call in operating system?
3. Why is context switching considered as an overhead to the system?
4. How is inter process communication implement using shared memory?
5. Describe resource allocation graph for the following. a) with a deadlock b)with a cycle but no deadlock.
6. What is critical section? What requirement should be satisfied by a solution to the critical section problem?
7. Consider the reference string 1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6. How many page faults occur while using FCFS for the following cases. a) frame=2 b)frame=3
8. Differentiate between internal and external fragmentations.
9. Compare sequential access and direct access methods of storage devices.
10. Define the terms (i) Disk bandwidth (ii) Seek time.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the following structures of operating system (i) Monolithic systems (ii) Layered Systems (iii) Micro Kernel (iv) Modular approach. (12)
- (b) Under what circumstances would a user be better off using a time sharing system than a PC or a single user workstation? (2)

OR

12. (a) What is the main advantage of the micro kernel approach to system design? How do user program and system program interact in a microkernel architecture? (8)
- (b) Describe the differences between symmetric and asymmetric multiprocessing? What are the advantages and disadvantages of multiprocessor systems? (6)
13. (a) Define process. With the help of a neat diagram explain different states of process. (8)
- (b) Explain how a new process can be created in Unix using fork system call. (6)

OR

14. (a) Find the average waiting time and average turnaround time for the processes given in the table below using:- i) SRT scheduling algorithm ii) Priority scheduling algorithm (9)

Process	Arrival Time (ms)	CPU Burst Time (ms)	Priority
P1	0	5	3
P2	2	4	1
P3	3	1	2
P4	5	2	4

- (b) What is a Process Control Block? Explain the fields used in a Process Control Block. (5)
15. (a) Consider a system with five processes P₀ through P₄ and three resources of type A, B, C. Resource type A has 10 instances, B has 5 instances and C has 7 instances. Suppose at time t₀ following snapshot of the system has been taken:

Process	Allocation			Max			Available		
	A	B	C	A	B	C	A	B	C
P ₀	0	1	0	7	5	3	3	3	2
P ₁	2	0	0	3	2	2			
P ₂	3	0	2	9	0	2			
P ₃	2	1	1	2	2	2			
P ₄	0	0	2	4	3	3			

- i. What will be the content of the Need matrix? Is the system in a safe state? If Yes, then what is the safe sequence? (8)

- ii. What will happen if process P1 requests one additional instance of resource type A and two instances of resource type C? (6)

OR

16. (a) State dining philosopher's problem and give a solution using semaphores. (7)
(b) What do you mean by binary semaphore and counting semaphore? With C struct, explain implementation of wait () and signal() (7)
17. (a) Consider the following page reference string 1,2,3,4,2,1,5,6,2,1,2,3,7,6,3,2,1,2,3,6. Find out the number of page faults if there are 4 page frames, using the following page replacement algorithms i) LRU ii) FIFO iii) Optimal (9)
(b) Explain the steps involved in handling a page fault. (5)

OR

18. (a) With a diagram, explain how paging is done with TLB. (5)
(b) Memory partitions of sizes 100 kb, 500 kb, 200 kb, 300 kb, 600 kb are available, how would best ,worst and first fit algorithms place processes of size 212 kb, 417 kb, 112 kb, 426 kb in order. Rank the algorithms in terms of how efficiently they uses memory. (9)
19. (a) Suppose that a disk drive has 5000 cylinders, numbered 0 to 4999. the drive currently services a request at cylinder 143, and the previous request was at cylinder 125. the queue of pending request in FIFO order is 86, 1470, 913, 1774, 948, 1509, 1022, 1750, 130. Starting from the current position, what is the total distance (in cylinders) that the disk arm moves to satisfy all pending requests for each of the following algorithms i) FCFS ii) SSFT iii) SCAN iv) LOOK v) C-SCAN (10)
(b) What is the use of access matrix in protection mechanism? (4)

OR

20. (a) Explain the different file allocation operations with advantages and disadvantages. (8)
(b) Explain the following i) file types ii) file operation iii) file attributes (6)

B24CS4G32	BASICS OF CLIENT SERVER COMPUTING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4		2024

Preamble

The primary purpose of this course is to introduce students to the foundational principles and evolving architectures of distributed client/server computing . The curriculum explores essential building blocks such as client/server hardware, network operating systems, middleware connectivity services like RPC and ODBC, and specialized database management architectures. It enables the students to design, implement, and manage secure and scalable client/server systems while effectively optimizing performance, reliability, and system administration across diverse network platforms.

Prerequisites

Nil

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Identify the fundamentals of client/server systems, server types, characteristics, and the driving forces behind their development. (Cognitive Knowledge Level: Understand)
CO 2	Explain client and server components, hardware platforms, operating systems, and communication using the OSI model. (Cognitive Knowledge Level: Understand)
CO 3	Apply middleware concepts, RPC mechanisms, and ODBC architecture to select appropriate connectivity solutions for distributed client/server applications. (Cognitive Knowledge Level: Apply)
CO 4	Apply system management techniques such as remote administration tools, performance monitoring, and scalability strategies to maintain efficient client/server systems. (Cognitive Knowledge Level: Apply)
CO 5	Apply client/server database architectures and middleware components to design suitable database-driven client/server solutions. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	2	1	2	1	2	2	1	1	1	1	1
CO 2	2	2	2	1	2	1	1	1	2	2	1
CO 3	1	3	2	2	2	2	2	2	2	2	1
CO 4	2	3	2	2	3	2	3	2	2	2	1
CO 5	1	3	2	2	1	2	3	3	2	2	1

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	40	40	40
Understand	40	40	40
Apply	20	20	20
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction to Client/Server Computing

Introduction to Client/Server computing - Basic Client/Server Computing Model, Server for Every Client-Server types- File Server, Print Server, Application Server, Mail Server, Directory Services Server, Web Server, Database Server, Transaction Servers. Characteristics -Fat or Thin, Stateless or Stateful, Servers and Mainframes, Client/Server Functions. Driving Forces behind Client/Server Computing- Business Perspective, Technology Perspective. Advantages and Disadvantages of Client/Server computing.

MODULE 2 (7 hours)

Client/Server Building Blocks

Components of Client Server Computing – Client – Role of the Client Client, Hardware (PC vs Workstation), Component of Client Applications–Server – Role of the Server, Server Hardware, Detailed Server Functionality , Connectivity – OSI – Communications Interface Technology. Operating Systems: Network Operating System (NOS), Server Operating Systems, Available Platforms (Windows/Linux/Unix concepts).

MODULE 3 (8 hours)

Middleware and Connectivity Services

Middleware – Definition, Types of Middleware , Specific Middleware – DCE, MOM, TP – Monitors. Connectivity Services-RPC (Remote Procedure Call): Concept and Usage. Request for Services, Base Services vs. External Services, Windows Services, Print Services, Remote Boot Services, Utility Services.

ODBC – Design Overview of ODBC – ODBC Architecture – Components – Applications – Driver Managers – Database Drivers – ODBC Data Sources

MODULE 4 (7 hours)

Client/ Server Systems Management and Support

Services and Support- System administration, Availability, Reliability, Scalability, Observability, Agility, Serviceability. Software Distribution, Performance, Network management. Remote Systems Management- RDP, Telnet, SSH, Security. LAN and Network Management issues.

MODULE 5 (7 hours)

Client/Server Databases

Client/Server Technology and Databases - Storing Data, Database System Architectures.

Client/Server in Respect of Databases- Client/Server Databases, Client/Server Database Computing, Database Computing Vs. Mainframe, PC/File Server Computing. Client/Server Database Architecture - Process-Per-Client Architecture, Multi-Threaded Architecture, Hybrid Architecture.

Database Middleware Component - Application Programming Interface, Database Translator.

Text Books

1. Patrick Smith & Steve Guengerich, Client / Server Computing, PHI
2. Subhash Chandra Yadav, Sanjay Kumar Singh, An Introduction to Client/Server Computing, New Age International Publishers
3. Alex Berson, Client/Server Architecture, McGraw Hill (Module 5)

Reference Books

1. Jeffrey D. Schank, "Novell's Guide to Client-Server Application & Architecture" Novell Press
2. Robert Orfali, Dan Harkey, Jeri Edwards, Client/Server Survival Guide, Wiley-India Edition, Third Edition
3. Dawna Travis Dewire, Client Server Computing — McGraw Hill
4. W.H. Inman, Developing Client Server Applications, BPB

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to Client/Server Computing		7 hours
1.1	Introduction to Client/Server computing - Basic Client/Server Computing Model	1
1.2	Server for Every Client-Server types- File Serve	1
1.3	Print Server, Application Server, Mail Server	1
1.4	Directory Services Server, Web Server, Database Server, Transaction Servers	1
1.5	Characteristics -Fat or Thin, Stateless or Stateful, Servers and Mainframes, Client/Server Functions	1

1.6	Driving Forces behind Client/Server Computing- Business Perspective, Technology Perspective	1
1.7	Advantages and Disadvantages of Client/Server Computing.	1
Module 2: Client/Server Building Blocks		7 hours
2.1	Components of Client Server Computing – Client – Role of the Client Client	1
2.2	Hardware (PC vs Workstation), Component of Client	1
2.3	Applications–Server – Role of the Server, Server Hardware	1
2.4	Detailed Server Functionality, Connectivity	1
2.5	SI – Communications Interface Technolog	1
2.6	Operating Systems: Network Operating System (NOS)	1
2.7	Server Operating Systems, Available Platforms (Windows/Linux/Unix concepts	1
Module 3: Middleware and Connectivity Services		8 hours
3.1	Middleware – Definition, Types of Middleware	1
3.2	Specific Middleware – DCE, MOM, TP – Monitors	1
3.3	Connectivity Services-RPC (Remote Procedure Call): Concept and Usage.	1
3.4	Request for Services, Base Services vs. External Services	1
3.5	Windows Services, Print Services, Remote Boot Services, Utility Services.	1
3.6	DBC – Design Overview of ODBC – ODBC Architecture	1
3.7	Components – Applications	1
3.8	Driver Managers – Database Drivers – ODBC Data Sources	1
Module 4: Client/ Server Systems Management and Support		7 hours
4.1	Services and Support- System administration,	1
4.2	Availability, Reliability, Scalability,	1
4.3	Observability, Agility, Serviceability	1
4.4	Software Distribution, Performance	1
4.5	Network management	1
4.6	Remote Systems Management- RDP, Telnet, SSH, Security	1
4.7	LAN and Network Management issues.	1
Module 5: Client/Server Databases		7 hours
5.1	Client/Server Technology and Databases - Storing Data, Database System Architectures.	1

5.2	Client/Server in Respect of Databases- Client/Server Databases	1
5.3	Client/Server Database Computing	1
5.4	Database Computing Vs. Mainframe, PC/File Server Computing.	1
5.5	Client/Server Database Architecture - Process-Per-Client Architecture	1
5.6	Multi-Threaded Architecture, Hybrid Architecture.	1
5.7	Database Middleware Component - Application Programming Interface, Database Translator.	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. A large manufacturing company has traditionally used a centralized mainframe system where all data processing and storage occur in one location, and branch offices only send requests and receive printed reports. Recently, the management decided to move toward a client/server architecture.

Explain how this shift would most likely change the day-to-day responsibilities of the IT staff at the branch offices compared to the mainframe era.

Identify one key business reason and one key technology reason that probably influenced this decision the most, and briefly explain why each factor would matter to this kind of organization.

2. A startup developing a mobile food-ordering app initially considered building a thick client application that would store most of the menu data and business logic directly on users' phones, but later chose a thin-client web-based approach with most processing happening on servers.

Describe what characteristic of client/server systems (fat vs. thin, stateless vs. stateful) is most clearly reflected in their final choice, and explain how this characteristic directly affects the user experience when using the app on different phones with varying storage and processing power.

Course Outcome 2 (CO 2):

1. A small accounting firm uses desktop PCs running Windows 10/11 for its employees and a dedicated server machine running Windows Server to store all client financial data and run the accounting software. Some employees complain that opening large reports is slow, especially when many people access the system at the same time.

Explain which part of the client/server building blocks (client hardware, server hardware, or connectivity) is most likely the primary bottleneck in this situation, and why the choice of

hardware platform for the server (compared to using a high-end workstation instead) makes a significant difference here.

2. A startup is setting up a new server and is debating between Linux and Windows Server OS. They require a platform that is highly customizable for web hosting but also need to ensure that their networking stack follows the OSI Model for compatibility with various third-party routers. Compare Linux and Windows in terms of "Available Platforms" and explain why the OSI model serves as a "common language" for communication regardless of which Operating System is chosen.

Course Outcome 3 (CO 3):

1. You are designing an asynchronous order processing system for a global e-commerce platform where the server might be busy during peak hours or occasionally offline. Would you implement this using Message-Oriented Middleware (MOM) or Remote Procedure Call (RPC)? Justify your selection by applying the concepts of synchronous vs. asynchronous communication and how your choice impacts system reliability.
2. A software firm is developing a data analytics tool that must simultaneously fetch records from an Oracle database and a MySQL database without writing separate code for each. Illustrate how you would apply the ODBC Architecture to solve this. Specifically, explain the role of the Driver Manager and Database Drivers in providing this database-independent connectivity.

Course Outcome 4 (CO 4):

1. A company's main server is located in a high-security data center in a different city. A system administrator needs to update the server configuration files on a headless Linux server while ensuring that the connection is encrypted to prevent data theft. Between SSH and Telnet, apply your knowledge of remote system management to select the correct tool. Explain the security implications of your choice in a LAN/WAN environment.
2. An online ticket booking application is experiencing crashes during the first hour of "Tatkal" bookings due to a sudden surge in traffic. Apply the concepts of Scalability and Availability to propose a management strategy. Would you focus on "Vertical Scaling" or "Horizontal Scaling" to ensure the system remains agile and serviceable during peak loads? Justify your approach.

Course Outcome 5 (CO 5):

1. You are building a high-performance banking database system that needs to handle thousands of simultaneous user transactions with minimal memory overhead. Evaluate the Process-Per-Client Architecture versus the Multi-Threaded Architecture. Select the most efficient architecture for this scenario and explain how your choice optimizes system resources compared to the alternative.
2. A hospital management system uses an application written in Java, but its legacy database uses a proprietary query format that the application does not understand. Apply your knowledge of Database Middleware Components to solve this. Explain how a Database Translator and an Application Programming Interface (API) would work together to allow the Java application to communicate with the legacy database effectively.

MODEL QUESTION PAPER

QP CODE:

Pages: 3

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

EIGHTH SEMESTER B.TECH DEGREE EXAMINATION, MAY, 2028

Course Code: B24CS4G32

Course Name: BASICS OF CLIENT SERVER COMPUTING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Describe any three types of servers used in Client-Server architecture.
2. Explain the difference between a fat client and a thin client.
3. Explain the importance of communication interfaces in networking.
4. Describe how a client application is structured.
5. What is Remote Procedure Call (RPC)?
6. Explain the function of Message-Oriented Middleware (MOM).
7. Explain how scalability improves system performance.
8. A system experiences frequent downtime due to hardware failures. Identify which property (availability or reliability) is affected and explain briefly.
9. Describe the role of an Application Programming Interface (API) in database access.
10. Explain how Client/Server database systems improve data management.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Enumerate the advantages and disadvantages of Client/Server computing. (7)
(b) Compare different types of servers and explain their roles with suitable examples. (7)

OR

12. (a) Discuss the differences between stateless and stateful servers, including their advantages and limitations. (7)
(b) Explain the working of the Client/Server model and how responsibilities are divided between client and server. (7)
13. (a) Explain the role of operating systems (Windows, Linux, Unix) in Client-Server environments. (7)
(b) Design a simple Client-Server system, specifying the roles of client, server, and connectivity components. (7)

OR

14. (a) Compare PCs and workstations in terms of performance and usage. (7)
(b) A banking system serves thousands of customers simultaneously. Each client sends requests such as balance inquiry, fund transfer, and transaction history to the central server. Explain how the server functionality should be designed to efficiently handle multiple client requests. Discuss relevant features such as concurrency, reliability, and security. (7)
15. (a) Explain the difference between applications, driver managers, and database drivers in ODBC. (7)
(b) Explain the role of various connectivity services in a Client-Server environment. (7)

OR

16. (a) An enterprise application needs to access multiple databases (Oracle, MySQL, SQL Server) using a single standardized interface. Explain how ODBC architecture can be applied in this scenario. Describe the role of each component. (7)
(b) Explain how different types of middleware (DCE, MOM, TP Monitors) support distributed applications. (7)
17. (a) Explain how availability, reliability, and scalability together ensure efficient system performance. (7)
(b) An organization allows employees to remotely manage servers. However, sensitive data is being transmitted over the network, and security is a major concern. Compare RDP, Telnet, and SSH, and recommend the most appropriate protocol for this scenario with justification. (7)

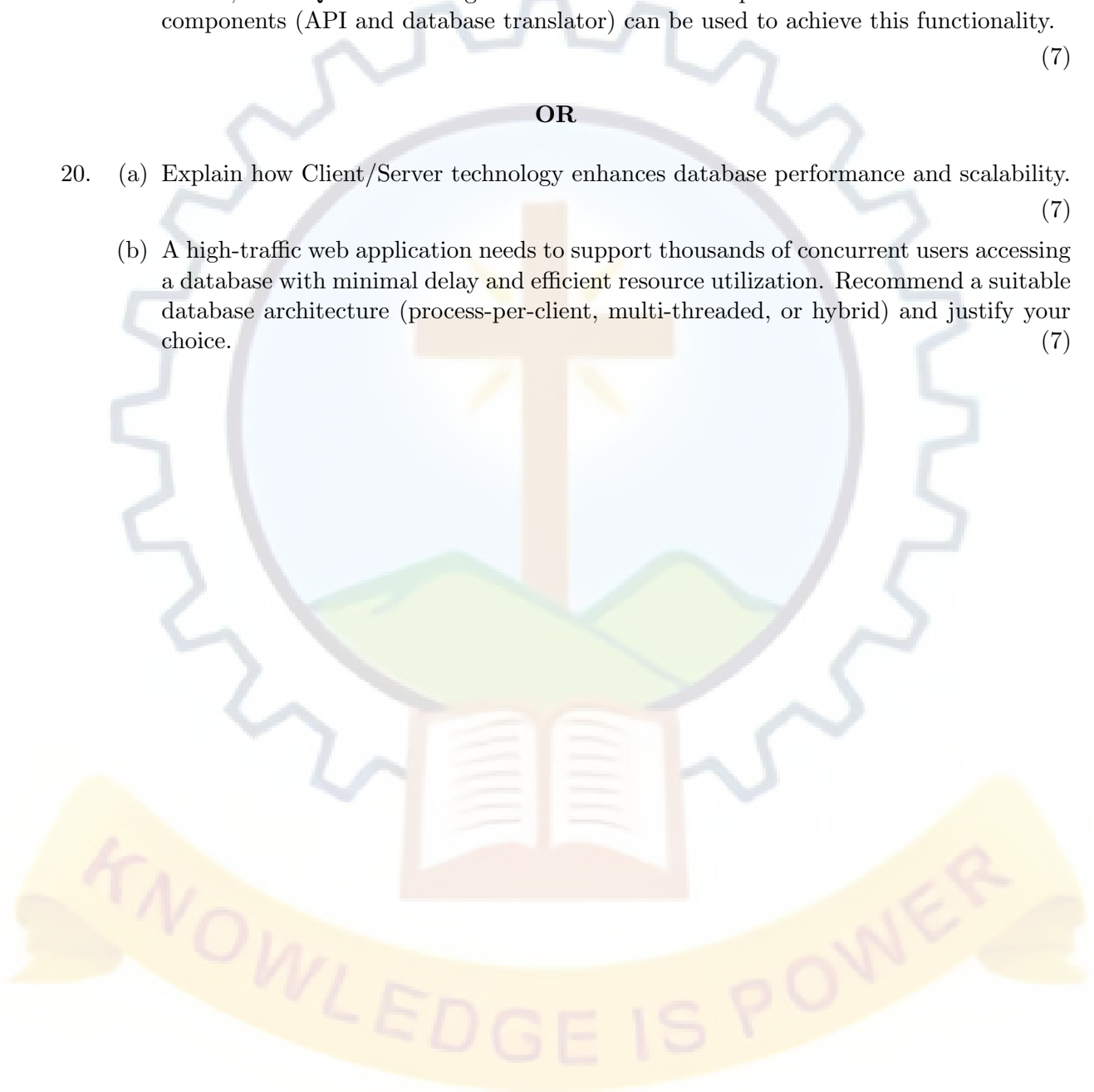
OR

18. (a) Discuss the importance of observability and serviceability in maintaining Client-Server systems. (7)

- (b) A multinational company needs to deploy software updates and patches across all its client systems efficiently and consistently. Explain how software distribution mechanisms can be used in this scenario and discuss their benefits. (7)
19. (a) Describe how Client/Server database systems handle data storage and retrieval efficiently. (7)
- (b) An enterprise application must interact with multiple database systems such as MySQL, Oracle, and SQL Server using a common interface. Explain how database middleware components (API and database translator) can be used to achieve this functionality. (7)

OR

20. (a) Explain how Client/Server technology enhances database performance and scalability. (7)
- (b) A high-traffic web application needs to support thousands of concurrent users accessing a database with minimal delay and efficient resource utilization. Recommend a suitable database architecture (process-per-client, multi-threaded, or hybrid) and justify your choice. (7)



B24CS4G33	INTRODUCTION TO DISTRIBUTED COMPUTING	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		2	1	0	4		2026

Preamble

The purpose of this course is to understand the system models, algorithms and protocols that allow computers to communicate and coordinate their actions to solve a problem. It covers communication and coordination mechanisms, techniques for ensuring consistency, replication, fault tolerance along with modern distributed computing platforms and their real-world applications. It helps the learners to develop solutions to problems in distributed computing environment.

Prerequisites

Basic knowledge in data structures and operating systems.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Summarize various aspects of distributed computation model and logical time (Cognitive Knowledge Level: Understand)
CO 2	Illustrate logical time, election algorithm, global snapshot algorithm and termination detection algorithm. (Cognitive Knowledge Level: Apply)
CO 3	Apply distributed mutual exclusion algorithms and deadlock detection techniques to ensure correct and efficient resource sharing in distributed systems. (Cognitive Knowledge Level: Apply)
CO 4	Recognize the significance of consistency and fault tolerance techniques in distributed systems. (Cognitive Knowledge Level: Apply)
CO 5	Understand modern distributed platforms and security (Cognitive Knowledge Level: Understand)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO 1	3	2	1	1	1	1	3	3	3		2
CO 2	3	2	2	2	3	1		1	2	1	3
CO 3	3	3	3	3	3	2	2	3	3	3	3
CO 4	3	3	3	3	3	3	3	3	3	3	3
CO 5	3	3	3	3	3	3	3	3	3	3	3

Assessment Pattern

Bloom's Category	Continuous Assessment		End Semester Examination (% Marks)
	Test 1 (%Marks)	Test 2 (%Marks)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance	10 marks
Continuous Assessment Test (2 numbers)	25 marks
Assignment/Quiz/Course Project	10 marks
Self study(quiz/seminar/micro project/ video assignments/ analytical assignments)	5 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contain 10 questions carrying 3 marks each. Part B contains 2 questions from each module out of which 1 to be answered. Each question carries 14 marks and can have maximum 2 sub divisions.

SYLLABUS

MODULE 1 (7 hours)

Introduction to Distributed Systems

Distributed System - Definition, Motivation, features of distributed systems, Design goals: transparency, openness, scalability, Primitives for distributed communication, Models of communication networks, Types of distributed systems, Architectural models: client-server, P2P, hybrid, Challenges: heterogeneity, security, scalability, Case studies of real-world distributed systems.

MODULE 2 (7 hours)

Election algorithm, Global state and Termination detection

Logical time – A framework for a system of logical clocks, Scalar time, Vector time. Leader election algorithm – Bully algorithm, Ring algorithm. Global state and snapshot recording algorithms – System model and definitions, Snapshot algorithm for FIFO channels – Chandy Lamport algorithm. Termination detection – System model of a distributed computation, Termination detection using distributed snapshots, Termination detection by weight throwing, Spanning-tree-based algorithm.

MODULE 3 (8 hours)

Mutual exclusion and Deadlock detection

Distributed mutual exclusion algorithms – System model, Requirements of mutual exclusion algorithm. Lamport's algorithm, Ricart-Agrawala algorithm, Quorum-based mutual exclusion algorithms – Maekawa's algorithm. Token-based algorithm – Suzuki-Kasami's broadcast algorithm. Deadlock detection in distributed systems – System model, Deadlock handling strategies, Issues in deadlock detection, Models of deadlocks.

MODULE 4 (7 hours)

Consistency, Replication and Fault Tolerance

Consistency models: strong, weak, eventual, Replication techniques, Distributed transactions, Two-Phase Commit protocol, Failure models and fault tolerance, Check pointing and rollback recovery – System model, consistent and inconsistent states, different types of messages, Issues in failure recovery, checkpoint based recovery.

MODULE 5 (7 hours)

Distributed file systems and Real-World Systems

Distributed file system – File service architecture, Case studies: Sun Network File System, Andrew File System, Google File System, HDFS, Distributed databases basics, Cloud computing fundamentals, Introduction to blockchain.

Textbooks

1. Ajay D. Kshemkalyani and Mukesh Singhal, Distributed Computing: Principles, Algorithms, and Systems, Cambridge University Press, 2011.
2. George Coulouris, Jean Dollimore, Tim Kindberg and Gordon Blair. Distributed Systems: Concepts and Design, Addison Wesley, Fifth edition.
3. Maarten Van Steen, Andrew S. Tanenbaum, Distributed Systems, Prentice Hall of India, Third edition, 2017.

Reference Books

1. Kai Hwang, Geoffrey C Fox, Jack J Dongarra, Distributed and Cloud Computing – From Parallel Processing to the Internet of Things, Morgan Kaufmann Publishers, 2012.
2. Sukumar Ghosh, Distributed Systems: An Algorithmic Approach, CRC Press, Second edition, 2015.
3. Randy Chow and Theodore Johnson, Distributed Operating Systems and Algorithm Analysis, Pearson Education India, First edition, 2009.
4. Valmir C. Barbosa, An Introduction to Distributed Algorithms, MIT Press, 2003.

COURSE CONTENTS AND LECTURE SCHEDULE

No.	Topic	No. of Lecture/ Tutorial Hours
Total Hours		36 Hours
Module 1: Introduction to Distributed systems		7 hours
1.1	Distributed System - Definition, Motivation, features of distributed systems	1
1.2	Primitives for distributed communication.	1
1.3	Design goals: transparency, openness, scalability	1
1.4	Models of communication networks,	1
1.5	Types of distributed systems, Architectural models: client-server, P2P, hybrid,	1
1.6	Challenges: heterogeneity, security, scalability,	1
1.7	Case studies of real-world distributed systems	1
Module 2: Election algorithm, Global state and Termination detection		7 hours
2.1	Logical time – A framework for a system of logical clocks, Scalar time	1
2.2	Vector time.	1

2.3	Leader election algorithm – Bully Algorithm, Ring Algorithm	1
2.4	Global state and snapshot recording algorithms – System model and definitions	1
2.5	Snapshot algorithm for FIFO channels – Chandy Lamport algorithm.	1
2.6	Termination detection – System model of a distributed computation, Termination detection using distributed snapshots	1
2.7	Termination detection by weight throwing, Spanning tree-based algorithm	1
Module 3: Mutual exclusion and Deadlock detection		8 hours
3.1	Distributed mutual exclusion algorithms – System model, Lamport's algorithm	1
3.2	Ricart–Agrawala algorithm	1
3.3	Quorum-based mutual exclusion algorithms – Maekawa's algorithm	1
3.4	Token-based algorithm – Suzuki–Kasami's broadcast algorithm.	1
3.5	Deadlock detection in distributed systems – System model, Deadlock handling strategies	2
3.6	Issues in deadlock detection	1
3.7	Models of deadlocks	1
Module 4: Consistency, Replication and Fault Tolerance		7 hours
4.1	Consistency models: strong, weak, eventual	1
4.2	Replication techniques	1
4.3	Two-Phase Commit protocol	1
4.4	Failure models and fault tolerance	1
4.5	Check pointing and rollback recovery – System model, consistent and inconsistent states, different types of messages	1
4.6	Issues in failure recovery	1
4.7	checkpoint based recovery	1
Module 5: Distributed file system and Real world systems		7 hours
5.1	Distributed file system – File service architecture,	1
5.2	Case Studies: Sun Network File System, Andrew File System,	1
5.3	Google File System.	1
5.4	HDFS	1
5.5	Distributed databases basics	1

5.6	Cloud computing fundamentals	1
5.7	Introduction to blockchain	1

CO ASSESSMENT QUESTIONS

Course Outcome 1 (CO1):

1. Describe different Architectural Models of distributed systems. Compare Client-Server, Peer-to-Peer (P2P), and Hybrid models.
2. Explain different forms of load balancing.

Course Outcome 2 (CO 2):

1. Apply ring-based leader election algorithm with 10 processes in the worst-performing case. Count the number of messages needed.
2. Apply spanning tree-based termination detection algorithm in the following scenario. The nodes are processes 0 to 6. Leaf nodes 3, 4, 5, and 6 are each given tokens T3, T4, T5 and T6 respectively. Leaf nodes 3, 4, 5 and 6 terminate in the order, but before terminating node 5, it sends a message to node 1.

Course Outcome 3 (CO 3):

1. What are the requirements of mutual exclusion algorithms?
2. Illustrate Suzuki-Kasami's broadcast algorithm.

Course Outcome 4 (CO 4):

1. Compare different models of deadlocks.
2. Explain the concept of Replication in distributed systems. What are the major benefits and challenges of replication?

Course Outcome 5 (CO 5):

1. Differentiate between Sun Network File System (NFS) and Andrew File System (AFS).
2. Explain the concepts of Fragmentation, Replication, and Allocation in Distributed Databases.

MODEL QUESTION PAPER

QP CODE:

Pages: 2

Reg.No.:

Name:

**MAR ATHANASIOUS COLLEGE OF ENGINEERING (AUTONOMOUS),
KOTHAMANGALAM**

EIGHTH SEMESTER B.TECH DEGREE EXAMINATION, MAY 2028

Course Code: B24CS4G33

Course Name: INTRODUCTION TO DISTRIBUTED COMPUTING

Max. Marks: 100

Duration: 3 hours

PART A

Answer all questions. Each question carries 3 marks.

1. Identify any three distributed applications and for each application, determine which all motivating factors are important for building an application over a distributed system.
2. Mention any three challenges in distributed systems.
3. Specify the issues in recording a global state.
4. Explain the rules used to update clocks in scalar time representation.
5. Describe how quorum-based mutual exclusion algorithms differ from the other categories of mutual exclusion algorithms.
6. Explain with example, how wait-for-graphs can be used in deadlock detection.
7. Why is Eventual Consistency widely used in modern distributed systems?
8. What are orphan messages and in-transit messages?
9. Define transparency in distributed systems. What are its different types?
10. Differentiate between whole file serving and whole file caching in Andrew file system.

PART B

Answer any one question from each module. Each question carries 14 marks.

11. (a) Explain the three different models of the service provided by communication networks. (9)
- (b) Explain how the causal dependency between events in distributed execution is defined using Lamport's happened-before relationship. (5)

OR

12. (a) Address the various strategies that can be adopted to satisfy the requirements of a reliable and fault tolerant distributed system. (5)
- (b) Which are the different versions of send and receive primitives for distributed communication? Explain. (9)
13. (a) Illustrate bully algorithm for electing a new leader. Does the algorithm meet liveness and safety conditions? (8)
- (b) Clearly mentioning assumptions, explain the rules of termination detection using distributed snapshots. (6)

OR

14. (a) In Chandy-Lamport algorithm for recording global snapshots, explain how the recorded local snapshots can be put together to create the global snapshot. Can multiple processes initiate the algorithm concurrently? (8)
- (b) Illustrate the working of spanning-tree-based termination detection algorithm. (6)
15. (a) Explain and illustrate Lamport's mutual exclusion algorithm. (6)
- (b) Discuss the three types of messages required for deadlock handling in Maekawa's algorithm. Explain how Maekawa's algorithm handles deadlocks. (8)

OR

16. (a) Explain and illustrate Ricart-Agrawala algorithm for achieving mutual exclusion. (8)
- (b) Explain any three different models of deadlock. (6)
17. (a) What are the issues in failure recovery? Illustrate with suitable examples. (7)
- (b) Explain why Eventual Consistency is preferred in large-scale web applications like social media and e-commerce. (7)

OR

18. (a) Differentiate consistent and inconsistent states with examples. (6)
- (b) What is checkpoint-based rollback-recovery? Explain the three classifications of checkpoint-based rollback-recovery. (8)
19. (a) Differentiate Andrew File System and NFS. (7)
- (b) Explain the architecture of Hadoop Distributed File System (HDFS) with a neat diagram. (7)

OR

20. (a) Explain the three service models of Cloud Computing (IaaS, PaaS, SaaS) with examples. (6)
- (b) Summarize distributed file system requirements. (8)

B24CS4L14	PROJECT PHASE II	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	12	3	6	2024

Preamble

The course 'Project Work Phase II' evokes innovation and invention skills in students. It offers an opportunity to synthesize and apply learned knowledge and analytical skills toward developing a functional prototype, simulation, or technology solution. Projects should align with thrust areas of the engineering stream or be interdisciplinary, prioritizing societal problems, sustainable development (aligned with UN SDGs such as SDG 7: Affordable & Clean Energy, SDG 9: Industry/Innovation/Infrastructure, SDG 13: Climate Action), and indigenous technologies.

Course Objectives

- To apply engineering knowledge in practical problem-solving and implementation.
- To foster innovation in design, development, and testing of sustainable engineering solutions.
- To develop teamwork, project management, and effective communication skills.

Course Outcomes (COs):

After successful completion, students will be able to:

CO1	Model, analyze, and solve real-world engineering problems by applying cross-domain knowledge. (Cognitive Knowledge Level:Apply)
CO2	Develop, test, and validate sustainable products/processes/technologies aligned with relevant UN SDGs. (Cognitive Knowledge Level:Apply)
CO3	Function effectively in teams, manage tasks/resources, and adhere to ethical/professional norms. (Cognitive Knowledge Level:Apply)
CO4	Identify research/technology gaps and propose innovative solutions with implementation evidence. (Cognitive Knowledge Level:Analyze)
CO5	Organize and communicate technical findings effectively through reports, presentations, and publications. Analyze

Mapping of Course Outcomes with Program Outcomes (POs)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	3	2	2	2	2	1	1	1	1	2
CO2	2	2	3	1	1	3	2	1	1	1	2
CO3					2	1	2	3	2	3	2
CO4	2	3	3	2	2	2	1	1		2	2
CO5					2	1	1	1	3	2	2

PROJECT PHASE II Targets

- In-depth study and refinement of the topic based on Phase-I report.
- Review and finalization of problem approach/methodology.
- Detailed action plan with timelines, resource allocation, and teamwork.
- Comprehensive analysis, modeling, simulation, design, experimentation, prototype development, testing, and validation.
- Final product/process development, results analysis, conclusions, and future directions (with SDG alignment where applicable).
- Preparation of conference/journal paper (if feasible).
- Presentation in university/state/national/international project expos.
- IPR filing (patent/publication) if applicable.
- Standard-format report preparation for departmental assessment.
- Final presentation and viva-voce with external expert.

Evaluation Guidelines & Rubrics

Total: **150 Marks**, Minimum to pass: **75 Marks**

Component	Marks
Project progress evaluation by guide	30
Two interim evaluations by Evaluation Committee	50
Final evaluation by Final Evaluation Committee	40
Quality of the report evaluated by committee	25
Paper publication / acceptance in conference or indexed journal	5
Total	150

Paper Publication / Acceptance (5 Marks): Awarded only if at least one paper (authored by the student team) is **published** or **accepted** in:

- A recognized national/international conference, or
- An indexed journal.
- Proof required: Acceptance letter / publication link / DOI / conference proceedings page.

Evaluation committee: HoD/senior faculty, Project coordinator, supervisor. **Final committee:** Project coordinator, industry/research/academic external expert, senior faculty from sister department.

Evaluation by the Guide (30 Marks)

- **Project Scheduling & Work Distribution (5):** Detailed timelines, Work Breakdown Structure (WBS), and balanced task allocation.
- **Literature Survey (4):** Demonstrates outstanding depth and breadth, updated from Phase-I.
- **Student's Diary / Daily Log (7):** Regular signed records of progress, observations, sketches, and reflections; promotes proper documentation and reasoning.
- **Individual Contribution (9):** Clear evidence of meaningful and equitable contribution from each student.
- **Project Completion / Demonstration (5):** Functional prototype or demonstration presented to the guide; objectives achieved and results verified.

Evaluation by the Guide (30 Marks)

- **Project Scheduling & Work Distribution (5):** Detailed timelines, Work Breakdown Structure (WBS), and balanced allocation.
- **Literature Survey (4):** Demonstrates outstanding depth and breadth, updated from Phase-I.
- **Student's Diary / Daily Log (7):** Regular signed records of progress, observations, sketches, and reflections; promotes documentation and reasoning.
- **Individual Contribution (9):** Evidence of meaningful and equitable input from each student.
- **Project Completion / Demonstration (5):** Functional prototype or demonstration presented to the guide; objectives met and results verified.

Rubrics for Evaluation by the Guide

Aspect / Parameter	Marks	Poor (0–1 for 5-mark items; 0–2 for 7–9 mark items)	Fair (2 for 5-mark; 3–4 for 7–9 mark)	Very Good (3–4 for 5-mark; 5–7 for 7–9 mark)	Outstanding (5 for 5-mark; 8–9 for 7–9 mark; full for others)
Project Scheduling & Distribution of Work among Team members [CO3, CO4 / PO8, PO10]	5	No meaningful scheduling or WBS; tasks unclear, unbalanced, or not assigned; no evidence of planning from Phase I.	Basic scheduling exists but incomplete/ ineffective; distribution skewed or vague; partial follow-up from Phase I plan.	Good detailed scheduling with timelines and WBS; distribution mostly balanced and supported by discussions/ journal; plan followed reasonably.	Excellent comprehensive scheduling, clear timelines, well-defined equitable WBS; strong documentation (e.g., tools/journal); all members actively engaged per capabilities.
Literature Survey [CO1, CO5 / PO1, PO2]	4	Minimal or superficial survey; lacks depth, relevance, or updates from Phase I; no critical analysis/gap identification.	Basic survey with some sources; limited depth or organization; partial updates but inadequate analysis.	Comprehensive survey with good range of sources; clear organization, relevant updates from Phase I; some gap analysis evident.	Outstanding in-depth, critical survey; broad/relevant sources; excellent synthesis, gap identification, and alignment with societal/SDG needs; fully refined from Phase I.
Student's Diary / Daily Log [CO4, CO5 / PO9, PO11]	7	Incomplete/ irregular entries; little/ no detail, sketches, reflections, or signatures; no evidence of documentation habit or reasoning.	Entries sporadic; basic observations but limited depth/ reflection; inconsistent signatures or detail.	Regular entries with good detail on progress, observations, sketches, suggestions; mostly signed; shows developing thought process.	Exemplary daily/ weekly records: detailed, reflective, with sketches/ drawings; consistent signatures; fosters strong reasoning, documentation, and lifelong learning habits.

Continued on next page

Aspect / Parameter	Marks	Poor	Fair	Very Good	Outstanding
Individual Contribution [CO3 / PO8]	9	Minimal/no meaningful participation by some/all members; contributions superficial or absent across stages.	Some contribution but uneven/limited; certain members dominate or under-contribute; superficial involvement.	Good individual input evident in most stages; contributions mostly equitable and documented (e.g., via journal/demo).	Outstanding equitable, substantial contributions by each member; clear leadership/ impact in key stages; well-evidenced via logs, demos, and team interaction.
Completion of the Project [CO1, CO2, CO5 / PO1-PO6]	5	No functional demonstration; objectives not met; results unverifiable or absent.	Partial demonstration; some objectives met but significant gaps; results limited/incomplete.	Good demonstration; most objectives achieved; results verifiable with minor issues.	Full functional demonstration (prototype/simulation); all objectives met; results validated, with clear evidence of innovation /sustainability/ SDG relevance.

Rubrics for Interim Evaluation – 1 (25 Marks)

No.	Parameter	Marks	Poor (0–1)	Fair (2–3)	Very Good (4)	Outstanding (5)
2-a	Novelty of idea & Implementation scope [CO5] (Group)	5	No useful societal/ SDG-relevant requirement addressed; lacks originality; routine implementation without fresh features.	Some aspects implementable; limited originality; minimal improvements over Phase-I.	Good implementable scope with evidence of originality; fresh specifications/features; independent learning/ ingenuity shown.	Highly novel/ original idea (unreported); ingenious, SDG-aligned implementation; fully implementable/ publishable quality.

2-b	Effectiveness of task distribution [CO3/PO8] (Group)	5	No distribution; members unclear on roles.	Allocated but ineffective/skewed; some members underutilized.	Good allocation via discussions/journal; mostly equitable/followed.	Excellent equitable distribution per capabilities; well-documented (journal/brainstorming); all actively contribute.
2-c	Adherence to project schedule [CO4/PO10] (Group)	5	Minimal planning; no adherence to Phase-I plan; unclear resources/budget; no journal.	Some progress over Phase-I; vague resources/budget; partial schedules.	Good planning/follow-up; resources listed; plan mostly followed; journal updated somewhat.	Detailed planning/tools (e.g., revision control); resource procurement on track; journal comprehensive; strong team interaction.
2-d	Interim Results & Progress [CO1/CO5] (Group)	5	No meaningful progress/results from Phase-I continuation.	Few results; limited analysis/inferences.	Several results achieved; issues identified with partial solutions.	Significant progress; clear inferences; SDG/ societal impact evident.
2-e	Presentation & Documentation [CO5/PO9] (Group/Individual)	5	Poor clarity/organization; no understanding shown.	Needs major improvement in structure/delivery.	Adequate content/delivery.	Professional, confident, well-structured presentation.

Rubrics for Interim Evaluation – 2 (25 Marks)

No.	Parameter	Marks	Poor (0-3)	Fair (4-6)	Very Good (7-9)	Outstanding (10)
2-f	Application of Engineering knowledge [CO1/ PO1-PO5] (Individual)	10	No evidence of knowledge application in implementation.	Basic application; inadequate for methodology.	Reasonable application in design/ testing.	Excellent, adaptive use of knowledge for robust solutions.
2-g	Involvement of individual members [CO3/PO8] (Individual)	5	Minimal/ no participation.	Superficial/ limited contribution.	Good involvement in core activities.	Leadership/ substantial technical impact.
2-h	Results and inferences [CO5/ PO2-PO4] (Group)	5	No outcomes achieved; no analysis.	Few outcomes; superficial inferences.	Many outcomes; good analysis/ issues addressed.	Comprehensive outcomes; deep inferences; SDG-aligned suggestions.
2-i	Documentation and presentation [CO5/PO9] (Individual/ Group)	5	Poor individual understanding/ delivery.	Needs improvement overall.	Satisfactory performance.	Professional, clear, confident delivery.

Rubrics for Final Evaluation (40 Marks)

No.	Parameter	Marks	Poor (0-3)	Fair (4-6)	Very Good (7-9)	Outstanding (10)
2-j	Engineering knowledge application [CO1/ PO1-PO5] (Group)	10	No effective application in final design/ execution.	Partial application; gaps in methodology.	Good application; minor improvements needed.	Outstanding integration for effective, validated solutions.
2-k	Relevance to societal/ industrial/ SDG needs [CO2/PO3/ PO6] (Group)	5	No clear relevance.	Some relevance; limited translation to specs.	Strong relevance; problem solved with societal/SDG considerations.	Exceptional impact; outstanding contribution to sustainable/ societal challenges.

No.	Parameter	Marks	Poor (0-3)	Fair (4-6)	Very Good (7-9)	Outstanding (10)
2-l	Innovation/ Novelty/ Creativity [CO5/PO3] (Group)	5	Routine; no originality.	Limited freshness.	Good ingenuity; implementable with improvements.	Highly novel/ ingenious; publishable/ IPR potential.
2-m	Quality of results/ conclusions/ solutions [CO1/CO5/ PO2/PO4] (Group)	10	None achieved; no valid conclusions.	Few achieved; limited analysis.	Many achieved; good inferences/ suggestions.	Most achieved; extensive validation; clear future directions/ SDG alignment.
2-n	Presentation/ Communication [CO5/PO9] (Group/ Individual)	10	Poor organization/ delivery.	Needs improvement.	Good flow/ professionalism.	Exceptional clarity/ confidence; engaging technical communication.

Rubrics for Report Evaluation (25 Marks)

No.	Parameter	Marks	Poor (0–9)	Fair (10–15)	Very Good (16–22)	Outstanding (23–25)
2-o	Report [CO5/PO9]	25	Shallow/ non- standard; poor organization; inadequate references; inconsistent/ unprofessional.	Partial standard format; language/ references need improvement; some inconsistencies.	Systematic/ standard format; good organization; consistent formatting; most references cited properly.	Exceptionally organized/ professional; all references/ diagrams/ tables correctly cited/ captioned; clear, readable, high- quality language; publication -ready quality.

B24CSML1	MINOR PROJECT	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	3	12	6	2024

Preamble

The course aims to assess students' ability to transform theoretical knowledge into a functional working model. It enables experience in organizing and implementing small-scale projects and facilitates the design and development of moderately complex systems (hardware or hardware-software combinations) with practical and sustainable applications.

Course Plan

In groups of up to 4 students, design and develop a moderately complex system addressing a real-world need. A functional working prototype/model is required, incorporating product design principles such as innovation, reliability, and aesthetics/ergonomics.

Students (in consultation with faculty/advisor):

- Identify topic of interest.
- Conduct literature review and gather information.
- State objectives and develop methodology.
- Perform design/fabrication/coding/programming.
- Demonstrate novelty/results/outputs (with SDG alignment where applicable).

Progress is evaluated through a minimum of two reviews by a committee. Students submit a project report and demonstrate the product against specifications. Emphasis is given to innovation, reliability, aesthetics/ergonomics, and societal/SDG relevance. **Course Outcomes**

After the completion of the course the student will be able to:

CO1	Apply acquired knowledge in the selected technology area for project development and implementation. (Cognitive Knowledge Level: Apply)
CO2	Identify, discuss, and justify technical/design aspects of the project systematically. (Cognitive Knowledge Level: Analyze)
CO3	Reproduce, improve, and refine technical aspects for engineering projects. (Cognitive Knowledge Level: Apply)
CO4	Collaborate effectively as a team in technical project development. (Cognitive Knowledge Level: Apply)
CO5	Communicate and report project activities and findings effectively (written/oral). (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	3	3	2	2	3	1	1	1	1	2
CO2	3	3	3	2	1	3	1	1	1	2	2
CO3	3	3	3	2	2	3	1	1	1	1	2
CO4					1	2	3	2	3	2	
CO5				1	1	1	2	3	2	2	

Evaluation Guidelines

Total: 150 marks (CIE 75 + ESE 75). Minimum pass as per university norms.

Split-up of CIE (75 Marks)

- Attendance : 10 Marks
- Marks awarded based on guide's evaluation : 15 Marks
- Project Report : 10 Marks
- Evaluation by Committee : 40 Marks

(Committee: 3 members – HoD/senior faculty, Academic coordinator/program coordinator, project guide/coordinator.)

Split-up of ESE (75 Marks) (1-hour duration)

- Level of completion : 10 Marks
- Demonstration of functionality/specifications : 25 Marks
- Project Report : 10 Marks
- Viva-voce : 20 Marks
- Presentation : 10 Marks

Enhanced rubrics for all evaluation components.

1. Guide's Evaluation (CIE – 15 Marks)

Holistic assessment by the guide based on regular monitoring, involvement, and progress.

Aspect	Marks Allocation	Poor (0–3)	Fair (4–7)	Good (8–11)	Very Good (12–14)	Outstanding (15)
Project Planning, Scheduling & Task Distribution [CO4/PO8, PO10]	4	No plan/ WBS; unbalanced/ unclear roles	Basic plan; skewed distribution	Reasonable plan/ timelines; mostly balanced	Detailed WBS/ timelines; equitable distribution documented	Comprehensive planning; excellent equitable allocation; strong team coordination
Technical Progress & Implementation [CO1, CO3 / PO1–PO5]	5	Minimal progress; major gaps in design/ fabrication	Partial implementation; basic functionality	Good progress; most objectives met partially	Strong implementation; refined technical aspects	Excellent execution; innovative / refined design with SDG relevance
Individual & Team Involvement [CO4 / PO8]	3	Low/ no participation by some members	Uneven contribution	Good involvement; fair equity	Consistent active participation	Outstanding collaboration; leadership evident
Documentation & Regular Reviews [CO5 / PO9, PO11]	3	Poor/ irregular records; no updates	Basic records; inconsistent	Good records; regular updates	Detailed logs/ reviews	Exemplary documentation; reflective & timely

2. Evaluation by Committee (CIE – 40 Marks) Panel assessment (reviews, interim progress, knowledge, involvement).

Aspect	Marks Allocation	Poor (0–10)	Fair (11–20)	Good (21–28)	Very Good (29–35)	Outstanding (36–40)
Technical Depth & Justification [CO1–CO3 / PO1–PO4]	15	Superficial; no justification/ gaps	Basic explanation; limited depth	Reasonable justification; some analysis	Strong technical discussion/ systematic approach	Excellent depth; critical analysis; innovative/ SDG-aligned
Innovation & Practical Relevance [CO1, CO3 / PO3, PO6]	10	No novelty; routine	Limited innovation	Some novelty/ practical aspects	Good innovation; practical/ SDG relevance	Highly innovative; strong societal/ SDG impact
Team Work & Involvement [CO4 / PO8]	8	Poor collaboration	Uneven involvement	Good team effort	Strong collaboration	Exemplary teamwork/ leadership
Communication & Presentation Readiness [CO5 / PO9]	7	Poor clarity	Needs improvement	Adequate	Clear & structured	Professional & confident

3. ESE Components – Enhanced Rubrics (75 Marks Total)

- **Level of Completion (10 Marks) [CO1–CO3]**

- Poor (0–2): Incomplete; major unfinished parts.
- Fair (3–5): Partially complete; significant gaps.
- Good (6–7): Mostly complete; minor issues.
- Very Good (8): Fully complete with good quality.
- Outstanding (9–10): Exceptionally complete; exceeds expectations (innovative/SDG features).

- **Demonstration of Functionality/Specifications (25 Marks) [CO1, CO3 / PO3]**

- Poor (0–6): Non-functional; fails specifications.
- Fair (7–12): Basic functionality; many failures.
- Good (13–18): Functional; meets most specifications with issues.
- Very Good (19–22): Reliable demonstration; meets all key specifications.
- Outstanding (23–25): Flawless; innovative features; reliable/SDG-aligned performance.

- **Project Report (10 Marks – CIE + ESE combined) [CO5 / PO9]**

- Poor (0–2): Poor organization; incomplete or missing sections.
- Fair (3–5): Basic format; weak language/references.
- Good (6–7): Systematic; mostly consistent.
- Very Good (8): Well-organized; good references.
- Outstanding (9–10): Professional; excellent structure and readability; proper citations with SDG discussion.

- **Viva-voce (20 Marks) [CO2, CO5 / PO2, PO9]**

- Poor (0–5): Unable to explain basics; no understanding.
- Fair (6–10): Limited answers; frequent prompting required.
- Good (11–14): Answers most questions reasonably.
- Very Good (15–17): Confident; good depth and explanation.
- Outstanding (18–20): Excellent articulation; integrates concepts; handles challenges fluently.

- **Presentation (10 Marks) [CO5 / PO9]**

- Poor (0–2): Poor delivery and structure.
- Fair (3–5): Needs major improvement.
- Good (6–7): Clear and logical.
- Very Good (8): Professional flow.
- Outstanding (9–10): Engaging and confident; excellent visuals and technical clarity.

B24CSHL1	HONOURS PROJECT	L	T	P	S	CREDIT	YEAR OF INTRODUCTION
		0	0	6	9	6	2024

Preamble

The Honours Project aims to assess and develop advanced capabilities in B.Tech Honours students by transforming deep theoretical knowledge into an innovative, functional system, model, prototype, process, software solution, or technology intervention with research depth and practical/societal impact. It enables hands-on experience in organizing, implementing, and leading moderately complex to advanced projects (hardware, software, simulation-based, experimental, process-oriented, or hybrid) relevant to the student's core engineering branch or interdisciplinary areas. Emphasis is placed on product/process design concepts, novelty, reliability, aesthetics/ergonomics/user-centric design (where applicable), sustainability (aligned with UN SDGs such as SDG 9: Industry, Innovation and Infrastructure, SDG 13: Climate Action, SDG 7: Affordable and Clean Energy, SDG 6: Clean Water and Sanitation, SDG 11: Sustainable Cities and Communities, or other relevant goals), and potential for publication, patent/IPR, or technology transfer. Students (individual or in groups of 2–4) consult faculty/guides to identify advanced, relevant topics, conduct a thorough literature review and gap analysis, define clear objectives and methodology, implement/develop/validate the solution through modeling, simulation, experimentation, prototyping, or deployment, analyze results rigorously, and demonstrate outcomes with evidence of innovation and real-world applicability.

Course Plan

- Identify an advanced core engineering topic in consultation with the faculty/advisor.
- Conduct an in-depth literature review and perform gap analysis.
- Define objectives and develop a systematic methodology (with SDG alignment where relevant).
- Design, fabricate, or develop hardware/software/code; implement, test, and refine the system.
- Demonstrate novelty, functionality, reliability, and aesthetics/ergonomics through results and outputs.
- Evaluate progress through a minimum of two reviews by the committee.
- Submit a comprehensive project report including full design specifications.
- Explore opportunities for conference/journal publication or IPR filing.

Course Outcomes

After the completion of the course the student will be able to:

CO 1	Apply and extend acquired knowledge in a selected core engineering branch/technology area for advanced Honours project development. (Cognitive Knowledge Level: Apply)
CO 2	Identify, discuss, justify, and systematically approach technical/design aspects of innovative projects with SDG/societal relevance. (Cognitive Knowledge Level: Analyze)
CO 3	Reproduce, improve, refine, and validate technical solutions for complex engineering projects. (Cognitive Knowledge Level: Apply/Analyze)
CO 4	Collaborate effectively as a team in the development and management of advanced technical projects. (Cognitive Knowledge Level: Apply)
CO 5	Communicate and report project activities, findings, and innovations effectively in both written and oral forms. (Cognitive Knowledge Level: Apply)

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO1	3	3	3	2	2	3	1	1	1	1	2
CO2	3	3	3	3	2	3	2	1	1	2	2
CO3	3	3	3	2	2	3	1		1	1	2
CO4					1	1	2	3	2	3	2
CO5					1		1	1	3	2	2

Evaluation Scheme

Total: 150 marks (CIE 75 + ESE 75; ESE duration 1 hour).

Evaluation by a 3-member committee (HoD/senior faculty, academic/ program coordinator, project guide/ coordinator). The committee assesses completion, functionality/specification demonstration, presentation, viva voce, knowledge, and level of involvement.

Split-up of CIE (75 Marks)

- Attendance : 10
- Marks awarded based on guide's evaluation : 15
- Project Report : 10
- Evaluation by Committee : 40

Split-up of ESE (75 Marks)

- Level of completion : 10
- Demonstration of functionality : 25
- Project Report : 10
- Viva-voce : 20
- Presentation : 10

Enhanced rubrics with progressive levels tied to COs/POs/SDGs are used, ensuring evidence-based and fair evaluation.

1. Guide's Evaluation (CIE – 15 Marks) Holistic assessment via regular monitoring, logs, contribution, progress.

Aspect	Marks Allocation	Poor (0–3)	Fair (4–7)	Good (8–11)	Very Good (12–14)	Outstanding (15)
Progress Scheduling / Task Distribution [CO4 / PO8, PO10]	4	No plan/adherence; unbalanced roles	Basic plan; skewed/ineffective	Reasonable scheduling; mostly balanced	Detailed WBS/timelines; equitable	Exemplary tools/planning; full SDG-aligned engagement
Innovation / Literature / SDG Relevance [CO2, CO3 / PO2, PO3, PO6]	5	Superficial; no novelty/ SDG link	Limited depth/ gaps	Good review; some innovation	Strong analysis; SDG-integrated novelty	Outstanding research; creative/sustainable impact
Individual/ Team Contribution & Diary/Log [CO4, CO5 / PO8, PO11]	6	Minimal participation/no records	Limited/ no superficial	Good input; regular logs	Substantial/ reflective	Leadership/ impact; exemplary documentation

2. Evaluation by Committee (CIE – 40 Marks) Based on interim reviews (progress, demo, knowledge, involvement).

Parameter	Marks	Poor (0–8)	Fair (9–16)	Good (17–24)	Very Good (25–32)	Outstanding (33–40)
Technical Depth / Justification / SDG Alignment [CO1–CO3 / PO1–PO6]	15	No justification	Basic analysis	Reasonable approach	Strong SDG link	Exceptional innovation
Teamwork / Involvement [CO4 / PO8]	10	No collaboration	Limited teamwork	Good teamwork	Effective collaboration	Leadership
Interim Progress / Functionality [CO3 / PO3, PO5]	15	No progress	Partial results	Good progress	Reliable system	Highly innovative/ validated

3. ESE – Level of Completion (10 Marks)

- Poor (0–2): Incomplete/ no model
- Fair (3–4): Basic with major gaps
- Good (5–6): Most features completed
- Very Good (7–8): Near complete
- Outstanding (9–10): Fully advanced; exceeds specifications/ SDG innovation

4. ESE – Demonstration of Functionality (25 Marks)

- Poor (0–5): Non-functional
- Fair (6–10): Partial/ unreliable
- Good (11–15): Functional basics
- Very Good (16–20): Reliable system
- Outstanding (21–25): Fully functional; innovative and SDG-aligned

5. ESE – Project Report (10 Marks)

- Poor (0–2): Poor/ no references
- Fair (3–4): Basic/ inconsistent
- Good (5–6): Systematic with references
- Very Good (7–8): Professional with SDG discussion
- Outstanding (9–10): Exemplary/ publishable quality

6. ESE – Viva-voce (20 Marks)

- Poor (0–4): No understanding
- Fair (5–8): Basic knowledge
- Good (9–12): Good grasp
- Very Good (13–16): Strong understanding
- Outstanding (17–20): Deep insight with SDG integration

7. ESE – Presentation (10 Marks)

- Poor (0–2): Unclear delivery
- Fair (3–4): Needs improvement
- Good (5–6): Clear and structured
- Very Good (7–8): Professional delivery
- Outstanding (9–10): Engaging with excellent communication