Sister Nivedita University

DG 1/2 New Town, Kolkata – 700156
www.snuniv.ac.in

OBE-RELATED ACADEMIC CURRICULUM

School of Science & Technology
Department of Computer Science

Syllabus
for
Master of Computer Applications (MCA)

Regulations (R24)

A Satyam Roychowdhury initiative

R24–25 Academic Session

VISION

To be a centre of excellence in computing, AI, and data-driven technologies, recognized for impactful research and innovation, while producing globally competent, ethical professionals and entrepreneurs who lead digital transformation across industries, and advancing interdisciplinary solutions that serve societal needs and support sustainable national and regional development.

MISSION

1. Deliver an outcome-based curriculum that blends strong computing fundamentals with emerging technologies through labs, projects, and real-world problem solving.
2. Foster a culture of research, innovation, and industry collaboration, entrepreneurship support, and modern infrastructure.
3. Nurture professionalism, ethics, inclusivity, effective communication, leadership, and lifelong learning to prepare graduates for responsible societal impact.

PROGRAM EDUCATIONAL OBJECTIVES (PEO)

PEO1:

Professional Competence: Graduates of the MCA program should demonstrate a high level of professional competence in various areas of computer science and application development. This includes proficiency in programming languages, software engineering principles, database management, networking, and other relevant domains. They should be capable of applying these skills to analyze, design, and develop innovative solutions to complex computing problems.

PEO2:

Career Advancement: Graduates of the Masters of Computer Application (MCA) program will demonstrate significant career advancement and progression in the field of computer science and related industries. They will excel in their professional roles, leveraging their comprehensive knowledge and skills acquired during the program to contribute effectively to the organizations they serve. This includes assuming leadership positions, undertaking challenging projects, and making meaningful contributions to the advancement of technology and innovation.

PEO3:

Continuous Learning and Adaptability: Graduates of the MCA program will exhibit a commitment to lifelong learning and adaptability in the dynamic field of computer science. They will possess the ability to stay updated with emerging technologies, methodologies, and best practices, enabling them to tackle new challenges and opportunities effectively. This includes actively engaging in professional development activities, pursuing further education or certifications, and adapting to evolving industry trends and demands.

PEO4:

Ethical and Responsible Citizenship: Graduates of the MCA program will demonstrate ethical and responsible citizenship in their professional endeavors and societal interactions. They will uphold high ethical standards, integrity, and professionalism in their work, adhering to ethical guidelines and principles in the development and implementation of technology solutions. Additionally, they will contribute positively to their communities by leveraging technology for social good, addressing societal challenges, and promoting inclusivity, diversity, and sustainability in their actions and decisions.

PROGRAM SPECIFIC OBJECTIVES (PSO)

PSO1:

Advanced Problem-Solving Skills: Graduates of the MCA program will demonstrate proficiency in applying advanced problem-solving techniques and methodologies to analyze complex computational problems, design innovative solutions, and implement them effectively in various domains of computer applications.

PSO2:

Specialized Software Development Competence: MCA graduates will exhibit specialized competence in software development by demonstrating the ability to architect, develop, and deploy software solutions that meet the specific needs and requirements of diverse industries, leveraging contemporary tools, technologies, and best practices.

PSO3:

Effective Project Management Abilities: Upon completion of the MCA program, students will possess effective project management abilities, including the skills to plan, organize, and execute software projects efficiently within given constraints, while effectively communicating with stakeholders and managing project teams to achieve desired outcomes.

PROGRAM OUTCOMES (PO)

PO1 – Computational Knowledge: Apply knowledge of computing fundamentals, specialization-specific computing concepts, mathematics, and relevant domain knowledge to abstract and conceptualize computing models for well-defined and complex problems and requirements.

PO2 – Problem Analysis: Identify, formulate, review research literature, and analyze complex computing problems to reach substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.

PO3 – Design / Development of Solutions: Design and evaluate solutions for complex computing problems and design and evaluate systems, components, or processes that meet specified needs, with appropriate consideration for public health and safety, as well as cultural, societal, and environmental factors.

PO4 – Conduct Investigations of Complex Computing Problems: Use research-based knowledge and research methods, including the design of experiments, analysis and interpretation of data, and synthesis of information, to investigate complex computing problems and provide valid conclusions.

PO5 – Modern Tool Usage: Create, select, adapt, and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an informed understanding of their capabilities and limitations.

PO6 – Professional Ethics: Understand and commit to professional ethics, cyber regulations, responsibilities, and norms of professional computing practice.

PO7 – Life-long Learning: Recognize the need for, and have the preparation and ability to engage in, independent and life-long learning for continual professional development in the context of rapid technological change.

PO8 – Project Management and Finance: Demonstrate knowledge and understanding of computing and management principles and apply these to one's own work, as an individual and as a leader or member in a team, to manage projects and function effectively in multidisciplinary environments, with due consideration to financial aspects.

PO9 – Communication Efficacy: Communicate effectively with the computing community and with society at large by being able to comprehend and write effective reports, design clear documentation, make effective presentations, and give and understand clear instructions related to complex computing activities.

PO10 – Societal and Environmental Concern: Understand and assess societal, environmental, health, safety, legal, and cultural issues in local and global contexts, and recognize the consequent responsibilities relevant to professional computing practice.

PO11 – Individual and Team Work: Function effectively as an individual, and as a member or leader in diverse teams and multidisciplinary environments.

PO12 – Innovation and Entrepreneurship: Identify timely opportunities and use innovation to pursue these opportunities, creating value and wealth for the betterment of the individual and society at large through entrepreneurial and intrapreneurial initiatives.

Credit Definition

Type	Duration (in Hour)	Credit
Lecture (L)	1	1
Tutorial (T)	1	1
Practical (P)	2	1
Sessional (S)	2	1

Total Credit Distribution

Semester	Credits						Credits/ Semester
	BS	AECC/HSM	USC/MUS	CC/PCC	DSE/PEC	SEC/PSE	
1	3	3	2	15	0	0	23
2	0	3	2	19	0	0	24
3	2	5	2	7	3	6	25
4	0	0	2	0	0	18	20
Credits/ Course	**5**	**11**	**8**	**41**	**3**	**24**	**92**

Category Definition

Definition of Category/ Type	Abbreviation
Basic Science	BS
Professional Core Course	CC/PCC
Discipline Specific Elective/ Professional Elective Course	DSE/PEC
Ability Enhancement Compulsory Courses/ Humanities & Social Science including Management	AECC/HSM
Skill Enhancement Course/ Program Specific Elective	SEC/PSE
Mandatory/ University Specified (Environmental Sc./ Induction Training/ Indian Constitution/ Foreign language)	USC/MUS

FIRST YEAR

SEMESTER-I

Sl. No.	Paper Name	Code	Category	Credit	Type			
					L	T	P	S
1	Computer Organization and Architecture		CC/PCC	3	2	1	0	0
2	Data Structures		CC/PCC	3	2	1	0	0
3	Database Management Systems		CC/PCC	3	2	1	0	0
4	Discrete Mathematics		BS	3	2	1	0	0
5	Communicative English		AECC/HSM	3	3	0	0	0
6	Foreign Language – I (German/ Spanish/ Japanese)		USC/MUS	2	2	0	0	0
Practical								
7	Computer Architecture and Organization Lab		CC/PCC	2	0	0	4	0
8	Data Structures Lab		CC/PCC	2	0	0	4	0
9	Database Management Systems Lab		CC/PCC	2	0	0	4	0
Total Credit=23								

SEMESTER-II

Sl. No.	Paper Name	Code	Category	Credit	Type			
					L	T	P	S
1	Software Engineering using UML		CC/PCC	3	2	1	0	0
2	Computer Networks		CC/PCC	3	2	1	0	0
3	Python Programming		CC/PCC	3	2	0	2	0
4	Operating Systems		CC/PCC	4	3	1	0	0
5	Management Information System		AECC/HSM	3	3	0	0	0
6	Foreign Language – II (German/ Spanish / Japanese)		USC/MUS	2	2	0	0	0
Practical								
7	Software Engineering using UML Lab		CC/PCC	2	0	0	4	0
8	Python Programming Lab		CC/PCC	2	0	0	4	0
9	Operating Systems Lab		CC/PCC	2	0	0	4	0
Total Credit=24								

SECOND YEAR

SEMESTER-III

Sl. No.	Paper Name	Code	Category	Credit	Type			
					L	T	P	S
1	Artificial Intelligence		CC/PCC	3	2	1	0	0
2	Theory of Computation		CC/PCC	3	2	1	0	0
3	Elective		DSE/PEC	3	0	0	0	0
4	Operation Research		BS	2	2	0	0	0
5	Accounting and Management Control		AECC/HSM	3	2	1	0	0
6	Customer Relationship Management using Salesforce		AECC/HSM	2	1	0	2	0
7	Foreign Language – III (German/ Spanish/ Japanese)		USC/MUS	2	2	0	0	0
8	Minor Project-I		SEC/PSE	6	0	0	0	12
Practical								
9	Artificial Intelligence Lab		CC/PCC	1	0	0	2	0
				Total Credit=25				

Elective

A. Data Warehousing and Data Mining
B. Compiler Design
C. Distributed Database System
D. AI and Neural Network
E. Cryptography and Network Security
F. Machine Learning
G. Internet of Things
H. Cloud Computing

SEMESTER-IV

Sl. No.	Paper Name	Code	Category	Credit	Type			
					L	T	P	S
1	Major Project		SEC/PSE	12	0	0	0	24
2	Grand Viva		SEC/PSE	6	0	0	0	0
3	Foreign Language – I (German/ Spanish/ Japanese)		USC/MUS	2	2	0	0	0
				Total Credit=20				

FIRST YEAR

SEMESTER-I

Sl. No.	Paper Name	Code	Category	Credit	Type			
					L	T	P	S
1	Computer Organization and Architecture		CC/PCC	3	2	1	0	0
2	Data Structures		CC/PCC	3	2	1	0	0
3	Database Management Systems		CC/PCC	3	2	1	0	0
4	Discrete Mathematics		BS	3	2	1	0	0
5	Communicative English		AECC/HSM	3	3	0	0	0
6	Foreign Language – I (German/ Spanish/ Japanese)		USC/MUS	2	2	0	0	0
Practical								
7	Computer Architecture and Organization Lab		CC/PCC	2	0	0	4	0
8	Data Structures Lab		CC/PCC	2	0	0	4	0
9	Database Management Systems Lab		CC/PCC	2	0	0	4	0
Total Credit=23								

COMPUTER ORGANIZATION AND ARCHITECTURE

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Computer Organization and Architecture	**Course Credit:** 03[2-1-0]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 1^{st}

➠ Learning Objectives: After completing this course, students will be able to:

1. Understand architectural concepts, functional units, and computer organization principles.
2. Explain data representation formats, computer arithmetic, and IEEE floating-point standards.
3. Analyze memory hierarchy, cache design, and I/O processing mechanisms in modern architectures.
4. Interpret instruction set design, addressing modes, and CPU organization.
5. Examine pipelining, ILP, GPU architecture, vector processors, and contemporary parallel architectures.

➠ Prerequisite: Students should have basic knowledge of computer fundamentals, understanding of number systems and basic programming constructs, familiarity with memory and process concepts.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Data Representation and Computer Arithmetic	6	17%
Module-II: Foundations of Computer Architecture	4	11%
Module-III: Memory and Input-output Organization	8	22%
Module-IV: Computer Instruction Set	8	22%
Module-V: Pipelining and Instruction-Level Parallelism	6	17%
Module-VI: Modern Technologies in Computer Architectures	4	11%

➠ Syllabus Outline: .....

Module I: Data Representation and Computer Arithmetic: [6L]

Number systems, Complement of numbers, Binary data representation (fixed point and floating point), Arithmetic operations on fixed point numbers, IEEE standards for floating point numbers, Booth's Algorithm.

Module II: Foundations of Digital Computer Organization: [4L]

Introduction to boolean algebra, Logic gates, Combinational circuits, Overview of computer systems, Organization vs Architecture, Functional units: Input, Output, Memory, ALU, Control Unit Revisiting basic organization: ALU, CU, Memory, I/O; BIOS.

Module III: Memory and Input-output Organization: [8L]

Memory parameters, Memory hierarchy, Registers-Types of registers, Cache Memory (L1, L2, L3 caches); Main memory: RAM(SRAM, DRAM, SDRAM), ROM. Secondary memory: Evolution of storage (Magnetic $\rightarrow$ Flash $\rightarrow$ NVMe $\rightarrow$ Cloud storage), Limitations of traditional HDD-based systems. Flash Memory Technologies: NAND, SSD; SSD vs HDD, NVMe Technology. Virtual Memory: Paging (including page replacement policy), Segmentation, TLB;

RAID and Redundant Storage Architectures: RAID 0, 1, 5, 6, 10; Hardware RAID vs Software RAID. Distributed and Cloud Storage Systems

Input-output devices and characteristics, Input-output processing, bus interface,data transfer techniques, I/O interrupts, channels.

Module IV: Computer Instruction Set: [8L]

Introduction to Instruction Sets, Instruction Format, CPU Organization, Instruction length, Addressing modes, Instruction Types: Data Transfer Instructions, Arithmetic Instructions, Logical Instructions, Control Instructions, Branching Instructions I/O Instructions; System Instructions: interrupts, traps, privileged instructions; RISC Vs. CISC

Module V: Pipelining and Instruction-Level Parallelism: [6L]

Performance measurement of computers: CPU Response time, execution time, Clock and CPI, Throughput, Latency, MIPS; Flynn's Classification: SISD, SIMD, MISD, MIMD Pipelining, performance of Pipeline processor (speed-up, efficiency, throughput), Types of Pipeline processor Pipeline hazards: Structural, Data, Control hazards; Advanced ILP: Superscalar, Superpipelining, VLIW Processors; NUMA and UMA memory models

Module VI: Modern Technologies in Computer Architectures: [4L]

GPU Architecture Basics, Vector Processors, Systolic Arrays (Overview), Introduction to Cloud and Server Architectures, Energy-Aware Architectures and Green Computing

GPU and GPGPU Architecture: CUDA architecture, GPU memory hierarchy, GPU vs CPU parallelism, Tensor cores in modern GPUs

➠ Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books:

Text Books:

1. Mano, M, "Computer System and Architecture", (3rd edition) Prentice Hall of India, New Delhi, 1994.
2. Pal Chauduri, P., "Computer Organization and Design", Prentice Hall of India, New Delhi, 1994.

Reference Books:

1. Rajaraman,V., and Radhakrishnan, T., "Introduction to Digital Computer Design" (4th edition). Prentice Hall of India, New Delhi, 1997.
2. Stallings. W, "Computer Organization and Architecture, (2nd edition) Prentice Hall of India, New Delhi

➠ Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Explain* the foundational concepts of computer architecture, functional units, and system organization.

CO2: *Apply* number systems, data representation formats, and arithmetic algorithms (including IEEE floating point and Booth's algorithm).

CO3: *Analyze* memory hierarchy, cache, main memory technologies, virtual memory, and I/O organization.

CO4: *Evaluate* CPU instruction sets, addressing modes, instruction formats, and compare RISC vs CISC architectures.

CO5: *Examine* pipelining, ILP, GPU/GPGPU architectures, vector/systolic processors, and modern computing paradigms.

➠ CO-PO-PSO Mapping: .....

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	-	1	-	-	-	-	-	-	-	2	1	-
CO2	3	3	1	-	2	-	-	-	-	-	-	-	3	2	-
CO3	3	3	2	1	3	-	-	-	-	-	-	-	3	3	2
CO4	3	3	3	1	2	-	-	-	-	-	-	-	3	3	2
CO5	2	3	3	2	3	-	1	-	-	1	1	2	3	3	3

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

DATA STRUCTURES

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Data Structures	**Course Credit:** 03[2-1-0]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 1^{st}

➠ Learning Objectives:

After completing this course, students will be able to:

1. Know the basic concepts of data structures and algorithms.
2. Understand concepts about searching and sorting techniques.
3. Understand basic concepts about stacks, queues, lists, trees and graphs.
4. Understanding about writing algorithms and step by step approach in
5. Solve problems with the help of fundamental data structure

➠ Prerequisite: Basics of programming language, Logic building skills.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Introduction to Data Structures	4	11%
Module-II: Structures and Arrays	4	11%
Module-III: Linear Data Structure	10	28%
Module-IV: Non-Linear Data Structure	10	28%
Module-V: Sorting Algorithms	4	11%
Module-VI: Searching and Hashing	4	11%

➠ Syllabus Outline: .....

Module I: Introduction to Data Structures: [4L]

Introduction, Definition, Classification of Data Structure, Description of Various Data Structures, Memory

Allocations in C, Algorithms, Algorithm Performance, Algorithm Analysis, Categories of Algorithms, Data Structure operations, Abstract Data Types.

Module II: Structures and Arrays: [4L]

Why Use Structures, Declaring a Structure, Accessing Structure Elements, How Structure Elements are Stored, Array of Structures, Additional Features of Structures, Uses of Structures.

Introduction, One Dimensional Array, Initializing One Dimensional Arrays, Accessing One Dimensional Arrays Elements, Implementation of One Dimensional Array in Memory, Passing Array to Functions, Insertion in One Dimensional Array, Deletion of Element One Dimensional Arrays, Traversing of an Array, Multi-Dimensional Arrays, Initializing a Two Dimensional Array, Accessing Two Dimensional Arrays Elements, Implementation of Two Dimensional Array in Memory, Pointers and Arrays, Array of Pointers, Array of Structures, Array within the Structure, Limitation of Linear Array.

Module III: Linear Data Structure: [10L]

Introduction, Stack Implementation, Operation on Stack, Stack Terminology, Algorithms for Push and Pop, Implementing Stack Using Pointers, Application of Stacks, Algorithm for Converting Infix to Expression to Postfix Form, Converting Infix to Expression to Prefix Form,Algorithm to Evaluate to Postfix Expression, Binary Expression Tree

Introduction, Implementation of Queue, Operation on a Queue, Algorithm for insertion and deletion on Queue (Using Array), Limitation of Simple Queue, Algorithm for insertion and deletion on Queue (Using Pointers), A Circular Queue, Double Ended Queues(deque), Priority Queue, Application of Queues, Multiple Queues

Introduction, Linked Lists, Key terms, Representation of linear linked list, Operations on linked list, Types of linkedlist, Singly linkedlist, Circular linkedlist, Doubly linkedlist, Circular doubly linkedlist, Application: Addition of two polynomials, Generalized Linkedlist,

Module IV: Non-Linear Data Structure: [10L]

Introduction, Tree, Binary trees, Binary trees representation, Creation of Binary tree, Operation on Binary trees, Technique of converting an expression into binary tree, Binary search tree, Threaded Binary Trees, Height balanced binary tree, B-Tree, B+Tree, Extended Binary tree Introduction, Defining graph, Basic terminology, Graph representation, Graph Implementation, Shortest path problem, Minimum Spanning tree, Shortest path algorithm.

Module V: Sorting Algorithms: [4L]

Introduction, Sorting, Bubble sort, Selection sort, Insertion sort, Quick sort.

Module VI: Searching and Hashing: [4L]

Introduction, Linear searching, Binary searching, Hashing, Terms associated with hash tables, Bucket overflow, Advantages of chaining.

➠ Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books:

Text Books:

1. Fundamentals of Data Structures in C, E.Horowitz- S.Sahni, Galgotia-2006
2. Data Structures and Algorithm Analysis in C, M.A.Weiss, Pearson Education-Fourth Edition

Reference Books:

1. Data Structures, Algorithms and Applications in C, Sartaj Sahni, University Press
2. Data Structures using C by Yedidyah Langsam, Moshe J. Augenstein and Aron M. Tananbaum, PHI.2002

➠ Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Explain* foundational data structure concepts, algorithm analysis, memory allocation, and abstract data types.

CO2: *Apply* structures, arrays, pointers, and ADTs to solve basic data organization and manipulation problems.

CO3: *Implement* and *analyze* linear data structures including stacks, queues, linked lists, and their applications.

CO4: *Analyze* and *evaluate* non-linear data structures such as trees, BSTs, AVL, B/B+ trees, and graphs for efficient data storage and retrieval.

CO5: *Apply* and *compare* searching, sorting, and hashing techniques to solve real-world computational problems.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	1	-	-	-	-	-	-	-	2	1	-
CO2	3	3	2	-	2	-	-	-	-	-	-	-	3	3	-
CO3	3	3	2	-	3	-	-	-	-	-	-	-	3	3	1
CO4	3	3	3	2	3	-	-	-	-	1	-	-	3	3	2
CO5	3	3	2	-	3	-	-	-	-	-	-	1	3	3	2

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

DATABASE MANAGEMENT SYSTEM

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Database Management System	**Course Credit:** 03[2-1-0]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 1^{st}

➠ Learning Objectives: After completing this course, students will be able to:

1. Understand the basic concepts and the applications of database systems
2. Be master of SQL and construct queries
3. Understand the relational database design principles
4. Become familiar with the basic issues of transaction processing and concurrency control
5. Become familiar with database storage structures and access techniques

➠ Prerequisite: Basic computer knowledge and knowledge about Data Structure and Algorithm

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Database System Architecture	4	11%
Module-II: Data Models	4	11%
Module-III: Database Design, ER-Diagram and Database Language	8	22%
Module-IV: Relational Algebra and Relational Calculus	8	22%
Module-V: Constraints, Views and SQL	6	17%
Module-VI: Indexing and Transactions	6	17%

➠ Syllabus Outline: .....

Module I: Database System Architecture: [4L]

Introduction, Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation

Language (DML).

Module II: Data Models: [4L]

Entity-relationship model, network model, relational and object-oriented data models, integrity constraints, data manipulation operations.

Module III: Database Design, ER-Diagram and Database Language: [8L]

Database design and ER Model: overview, ER-Model, Constraints, ER-Diagrams, ERD Issues, weak entity sets, Codd's rules, Relational Schemas, Introduction to UML, Relational database model: Logical view of data, keys, integrity rules. Relational Database design: features of good relational database design, atomic domain and Normalization (1NF, 2NF, 3NF, BCNF, 4NF).

Module IV: Relational Algebra and Relational Calculus: [8L]

Relational algebra: introduction, Selection and projection, set operations, renaming, Joins, Division, syntax, semantics. Operators, grouping and ungrouping, relational comparison. Calculus: Tuple relational calculus, Domain relational Calculus, calculus vs algebra, computational capabilities.

Module V: Constraints, Views and SQL: [6L]

What is constraints, types of constrains, Integrity constraints, Views: Introduction to views, data independence, security, updates on views, comparison between tables and views SQL: data definition, aggregate function, Null Values, nested sub queries, Joined relations. Triggers.

Module VI: Indexing and Transactions: [6L]

Indices, B-trees, B+ trees, hashing, Concept of transaction, Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp-based schedulers, Concurrency Control schemes, Database recovery.

➠ Teaching–Learning Methodology: .....

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books: .....

Text Books:

1. Data base System Concepts, Silberschatz, Korth, McGraw hill, Sixth Edition.
2. Fundamentals of Database Systems, Elmasri Navathe Pearson Education.

Reference Books:

1. An Introduction to Database systems, C.J. Date, A.Kannan, S.Swami Nadhan, Pearson, Eight Edition.
2. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, TATA McGrawHill 3rd Edition.

Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Explain* key database concepts, architecture, and system components to demonstrate conceptual understanding of database systems.

CO2: *Analyze* real-world requirements and develop conceptual data models with appropriate constraints and relationships.

CO3: *Construct* normalized relational database schemas by applying normalization techniques to eliminate redundancy and improve data integrity.

CO4: *Apply* relational algebra and SQL operations to query, manipulate, and manage data in relational database systems under defined constraints.

CO5: *Evaluate* transaction management, concurrency control, and indexing mechanisms to ensure reliable, consistent, and efficient database operations.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	1	-	-	-	1	-	-	2	-	-
CO2	2	3	3	-	-	-	-	-	-	-	-	-	3	2	-
CO3	2	2	3	-	2	-	-	-	-	-	-	-	3	3	-
CO4	2	3	-	2	3	-	-	-	1	-	-	-	3	3	2
CO5	2	2	2	2	3	1	1	-	-	1	-	-	3	3	2

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

DISCRETE MATHEMATICS

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Discrete Mathematics	**Course Credit:** 03[2-1-0]
Department: Computer Science Engineering	**Category:** BS
Code: XXXXXX	**Semester:** 1^{st}

➠ Learning Objectives: On completion of the course, student will be able to

1. Understand the fundamentals of Boolean algebra, logic gates, canonical forms, and Karnaugh maps, and apply them to logical simplification and digital system design.
2. Explain and apply core concepts of abstract algebra to model discrete structures.
3. Apply the knowledge of graph theory to solve complex engineering problem.

➠ Prerequisite:

Before learning the concepts of Discrete Mathematics, you should have a basic knowledge of set, relation, mapping, matrix etc.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Boolean Algebra	6	20%
Module-II: Abstract Algebra	6	13%
Module-III: Combinatories	6	13%
Module-IV: Fundamental concepts of Graph Theory	6	13%
Module-V: Tree and Net workflow	6	24%
Module-VI: Logic	6	17%

➠ Syllabus Outline: .....

Module I: Boolean Algebra: [6L]

Introduction of Boolean algebra, truth table, basic logic gate, basic postulates of Boolean algebra,

principle of duality, canonical form, Karnaugh map.

Module II: Abstract Algebra: [6L]

Set, Functions, relation, partially ordered sets, lattice, distributive and complete lattices, group, ring, field.

Module III: Combinatories: [6L]

Pascal Triangle, Basic counting, balls and bins problems, generating functions, recurrence relations. Principle of mathematical induction, pigeonhole principle. Principle of inclusion and exclusion.

Module IV: Fundamental concepts of Graph Theory: [6L]

Graphs and digraphs, complement, isomorphism, connectedness and reachability, adjacency matrix, Eulerian paths and circuits in graphs and digraphs, Hamiltonian paths and circuits in graphs and tournaments,Planar graphs, Euler's formula, dual of a planer graph, independence number and clique number, chromatic number, statement of Four-color theorem.

Module V: Tree and Net workflow: [6L]

Basics: equivalent characterizations of trees, forests, Spanning trees and 2-switches, Distance and center, Optimization: Kruskal's Theorem and Dijkstra's Theorem. Networkflow, Max-flowMin-cuttheorem(statementonly); Ford-Fulkerson algorithm.

Module VI: Logic: [6L]

Propositional calculus - propositions and connectives, syntax; Semantics - truth assignments and truth tables, validity and satisfiability, tautology; Adequate set of connectives; Equivalence and normal forms; Compactness and resolution; Formal reducibility - natural deduction system and axiom system;Soundness and completeness. Distributive and complete lattices.

➠ Teaching–Learning Methodology: .....

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books: .....

Text Books:

1. Topics in Algebra, I. N. Herstein, John Wiley and Sons.

2. Digital Logic & Computer Design, M. Morris Mano, Pearson.
3. Elements of Discrete Mathematics, (Second Edition) C. L. LiuMcGraw Hill, New Delhi.
4. Graph Theory with Applications, J. A. Bondy and U. S. R. Murty, Macmillan Press, London.
5. Mathematical Logic for Computer Science, L. Zhongwan, World Scientific, Singapore.

Reference Books:

1. Introduction to linear algebra. Gilbert Strang.
2. Introductory Combinatorics, R. A. Brualdi, North-Holland, New York.
3. Graph Theory with Applications to Engineering and Computer Science, N. Deo, Prentice Hall, Englewood Cliffs.
4. Introduction to Mathematical Logic, (Second Edition), E. Mendelsohn, Van-Nostrand, London.

Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Explain* the fundamental properties to solve problems in algebraic structures with clarity.

CO2: *Solve* logical structures and standard forms accurately.

CO3: *Examine* analytical techniques to solve mathematical or computational problems.

CO4: *Design* efficient systems using logical and systematic methods.

CO5: *Evaluate* complex structures to determine optimal solutions.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	1	-	-	1	1	1	-	2	2	1
CO2	3	3	1	1	2	1	-	-	2	1	1	-	3	3	1
CO3	3	3	2	2	2	-	-	1	2	1	1	-	3	3	2
CO4	3	3	3	2	2	1	-	2	2	2	2	1	3	3	2
CO5	3	3	3	3	2	1	-	2	2	2	3	1	3	3	2

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

COMMUNICATIVE ENGLISH

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Communicative English	**Course Credit:** 03[3-0-0]
Department: Computer Science Engineering	**Category:** AECC/HSM
Code: XXXXXX	**Semester:** 1^{st}

➠ Learning Objectives:

After completing this course, students will be able to:

1. To enhance the level of literary and aesthetic experience of students and to help them respond creatively.
2. To sensitize students to the major issues in the society and the world.
3. To provide the students with an ability to build and enrich their communication skills.
4. To equip students to utilize the digital knowledge resources effectively for their chosen fields of study.
5. To help them think and write imaginatively and critically.
6. To broaden their outlook and sensibility and acquaint them with cultural diversity and divergence in perspectives.
7. Equip them with basic knowledge to pursue careers in publishing, cinema, theatre, journalism, education and advertising.

➠ Prerequisite:

Fundamental knowledge of English and Grammar.

Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Essay and precis writing	6	16%
Module-II: Slide preparation and oral presentation principles	6	16%
Module-III: Written presentation of technical material	6	17%
Module-IV: Preparation of bibliography, basic of official correspondence	6	17%
Module-V: Preparation of bio-data	6	17 %
Module-VI: Group discussions should also be used and feedback is given to students	6	17%

Syllabus Outline:

Module I: Essay and precis writing: [6L]

Essay and precis writing

Module II: Slide preparation and oral presentation principles: [6L]

Slide preparation and oral presentation principles

Module III: Written presentation of technical material: [6L]

Written presentation of technical material

Module IV: Preparation of bibliography, basic of official correspondence: [6L]

Preparation of bibliography, basic of official correspondence

Module V: Preparation of bio-data: [6L]

Preparation of bio-data

Module VI: Group discussions should also be used and feedback is given to students: [6L]

Group discussions should also be used and feedback is given to students

Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

Text & Reference Books:

Text Books:

1. The Chicago Manual of Style, 13th Edition, Prentice Hall of India, 1989 Gowers, Ernest, ”The Complete Words”. Penguin, 1973.
2. IEEE Transactions on ”Written and Oral Communications” has many papers of relevance 3. Ludlow, R., and Panton, F., ”The Essence of Effective Communication”, Prentice Hall of India Pvt.Ltd. 1995.

Reference Books:

1. Menzel, D.H., Jones, H.M., Boyd, L.G., ”Writing a Technical Paper”. McGraw Hill, 1961.
2. Strunk, W., White. E.B., ”The Elements of Style”, 3rd Edition, McMillan, 1979.
3. Munter, M., ”Business Communication: Strategy and Prentice Hall, New Jersey, 1987.
4. Tubian, K.L., ”A Manual for Writen of 1erm Papers, Thesis and Dissertation”, Univ. of Chicago Press, 1973.

Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Write* clear, structured essays and précis with appropriate vocabulary and coherence.

CO2: *Prepare* professional slides and deliver effective oral presentations.

CO3: *Compose* well-structured technical documents with clarity and correctness.

CO4: *Prepare* formal bibliographies and official correspondence following professional standards.

CO5: *Demonstrate* effective communication skills in group discussions with constructive feedback.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	-	-	-	-	-	-	-	1	3	-	1	-	-	-	1
CO2	-	-	-	-	-	-	-	2	3	-	2	-	-	-	2
CO3	-	2	-	-	-	-	-	1	3	-	1	-	-	-	2
CO4	-	1	-	-	-	-	-	1	3	-	1	-	-	-	1
CO5	-	-	-	-	-	-	-	2	3	-	3	-	-	-	2

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

COMPUTER ARCHITECTURE AND ORGANIZATION LAB

Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Computer Architecture and Organization Lab	**Course Credit:** 02[0-0-4]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 1^{st}

List of Practicals:

1. To design the circuit of half adder.
2. To design the circuit of full adder.
3. To design the circuit of half subtractor.
4. To design the circuit of full subtractor.
5. To design an 8×1 Multiplexer.
6. To design a 4 bit combinational shifter.
7. To design a BCD adder
8. To design a 4-bit adder subtractor.
9. To design 2:4 Decoder
10. To design an ALU.
11. Simulation of simple fundamental units like half adder, full adder, multiplexer, de-multiplexer, Arithmetic logic Unit, Simple processor (CPU) etc using VHDL code. (Using Xilinx)

DATA STRUCTURES LAB

Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Data Structures Lab	**Course Credit:** 02[0-0-4]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 1st

List of Practicals: .....

1. Implement a stack by using array then do the PUSH & POP operation
2. Write a program to evaluate a postfix notation.
3. Write a program to convert infix to postfix.
4. Implement a Circular Queue by using array then do the enqueue and dqueue operation.
5. Implement Single Linked List and does insertion, deletion, display, reverse.
6. Implement Doubly Linked List and does insertion, deletion, display, reverse.
7. Implement a stack using linked lists.
8. Implement Circular Linked List, queue using linked lists.
9. Implement JOSEPHUS problem.
10. Write a program to add two polynomials.
11. Write a program to multiply two polynomials.
12. Write a program for addition of sparse matrix.
13. Write a program to multiplication of sparse matrix.
14. Create binary search tree and implement Preorder, Inorder, Postorder and delete an element from the tree
15. Implement a threaded binary tree and perform the inorder traversal operation.
16. Implement AVL tree.
17. Implement Splay tree.
18. Implement Priority Queue using Heap.
19. Implement BFS, DFS.
20. Implement Prim's and Kruskal's Algorithm.
21. Write a program to sort an array using Bubble sort.
22. Write a program to sort an array using Insertion sort
23. Write a program to sort an array using Selection sort.
24. Write a program to sort an array using Quick sort.

25. Write a program to sort an array using Merge sort.
26. Write a program to sort an array using Heap sort.
27. Write a program to sort an array using Radix sort.
28. Write a program to sort an array using Shell sort.
29. Implement Linear and Binary search.
30. Implement interpolation search.

DATABASE MANAGEMENT SYSTEM LAB

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Database Management System Lab	**Course Credit:** 02[0-0-4]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 1^{st}

➠ List of Practicals: .....

Introduction to SQL:

- Basic concepts of databases and DBMS
- Introduction to Structured Query Language (SQL)
- Creating, querying, updating, and deleting tables using SQL

Data Definition Language (DDL):

1. Creating and modifying database schema using DDL commands (CREATE, ALTER, DROP)
2. Constraints (PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL, CHECK)

Data Manipulation Language (DML):

1. Inserting, updating, and deleting data using DML commands (INSERT, UPDATE, DELETE)
2. Retrieving data using SELECT statement
3. Filtering data using WHERE clause
4. Sorting data using ORDER BY clause
5. Aggregating data using GROUP BY and aggregate functions (SUM, AVG, COUNT, MAX, MIN)

Joins and Subqueries:

1. Performing joins (INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN)
2. Writing subqueries to retrieve data
3. Understanding correlated subqueries

Indexes and Views:

1. Creating indexes for efficient data retrieval
2. Creating and managing views

3. Understanding materialized views

Transactions and Concurrency Control:

1. Introduction to transactions
2. ACID properties of transactions
3. Isolation levels (READ UNCOMMITTED, READ COMMITTED, REPEATABLE READ, SERIALIZABLE)
4. Locking mechanisms for concurrency control

Database Connectivity: Connecting to databases using programming languages (e.g., Java, Python) and APIs (e.g., JDBC, SQLAlchemy) Performing CRUD operations through programming languages

Database Administration:

1. Managing users and permissions
2. Backup and recovery strategies
3. Monitoring database performance
4. Tuning SQL queries for better performance

Normalization:

1. Understanding normalization forms (1NF, 2NF, 3NF, BCNF)
2. Applying normalization techniques to improve database design

Stored Procedures and Triggers:

1. Creating and executing stored procedures
2. Defining and executing triggers

Database Design Project: Students may be assigned a database design project where they have to design a database schema, implement it using SQL, and develop a simple application to interact with the database.

SEMESTER-II

Sl. No.	Paper Name	Code	Category	Credit	Type			
					L	T	P	S
1	Software Engineering using UML		CC/PCC	3	2	1	0	0
2	Computer Networks		CC/PCC	3	2	1	0	0
3	Python Programming		CC/PCC	3	2	0	2	0
4	Operating Systems		CC/PCC	4	3	1	0	0
5	Management Information System		AECC/HSM	3	3	0	0	0
6	Foreign Language – II (German/ Spanish / Japanese)		USC/MUS	2	2	0	0	0
Practical								
7	Software Engineering using UML Lab		CC/PCC	2	0	0	4	0
8	Python Programming Lab		CC/PCC	2	0	0	4	0
9	Operating Systems Lab		CC/PCC	2	0	0	4	0
Total Credit=24								

SOFTWARE ENGINEERING USING UML

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Software Engineering using UML	**Course Credit:** 03[2-1-0]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 2^{nd}

➠ Learning Objectives: After completing this course, students will be able to:

1. To capture the requirements specification for an intended software system.
2. To draw the UML diagrams for the given specification
3. To map the design properly to code
4. To test the software system thoroughly for all scenarios
5. To improve the design by applying appropriate design patterns

➠ Prerequisite: The Software Development Life Cycle is must and basic Analysis towards any System.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Introduction to on Object Oriented Technologies and the UML Method, Introduction to the UML Language	4	11%
Module-II: Requirements Analysis Using Case Modeling	8	22%
Module-III: Transfer from Analysis to Design in the Characterization Stage, Finding objects from Flow of Events	8	22%
Module-IV: The Logical View Design Stage: The Static Structure Diagrams	8	23%
Module-V: Package Diagram Model and Dynamic Model: State Diagram / Activity Diagram	4	11%
Module-VI: Component Diagram Model and Deployment Model	4	11%

➠ Syllabus Outline: .....

Module I: Introduction to on Object Oriented Technologies and the UML Method, Introduction to the UML Language: [4L]

Software development process: The Waterfall Model vs. The Spiral Model, The Software Crisis, description of the real world using the Objects Model, Classes, inheritance and multiple configurations, Quality software characteristics, Description of the Object-Oriented Analysis process vs. the Structure Analysis Model. Standards, Elements of the language, General description of various models, The process of Object-Oriented software development, Description of Design Patterns, Technological Description of Distributed Systems.

Module II: Requirements Analysis Using Case Modeling: [8L]

Analysis of system requirements, Actor definitions, writing a case goal, Use Case Diagrams, Use Case Relationships.

Module III: Transfer from Analysis to Design in the Characterization Stage, Finding objects from Flow of Events: [8L]

Interaction Diagrams, Description of goal, Defining UML Method, Operation, Object Interface, Class, Sequence Diagram. Describing the process of finding objects using a Sequence Diagram, Describing the process of finding objects using a Collaboration Diagram.

Module IV: The Logical View Design Stage: The Static Structure Diagrams: [8L]

The Class Diagram Model, Attributes descriptions, Operations descriptions, Connections descriptions in the Static Model, Association, Generalization, Aggregation, Dependency, Interfacing, Multiplicity.

Module V: Package Diagram Model and Dynamic Model: State Diagram / Activity Diagram: [4L]

Description of the model, White box, black box, Connections between packagers, Interfaces, Create Package Diagram, Drill Down. Description of the State Diagram, Events Handling, Description of the Activity Diagram, Exercise in State Machines.

Module VI: Component Diagram Model and Deployment Model: [4L]

Physical Aspect, Logical Aspect, Connections and Dependencies, User face, Initial DB design in a UML environment, Processors, Connections, Components, Tasks, Threads, Signals and Events.

➠ Teaching–Learning Methodology: .....

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

Text & Reference Books:

Text Books:

1. Object-Oriented Software Engineering: using UML, Patterns, and Java. Bernd Bruegge and Allen H. Dutoit.

Reference Books:

1. Design Patterns: Elements of Reusable Object-Oriented Software. Erich Gamma, Richard Helm, Ralph Johnson, and John M. Vlissides.
2. Bennett, S., McRobb, S., & Farmer, R. (2010). Object-Oriented Systems Analysis and Design Using UML (4th ed.). McGraw-Hill Education.
3. Pressman, R. S., & Maxim, B. R. (2020). Software Engineering: A Practitioner's Approach (9th ed.). McGraw-Hill Education

Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Explain* the fundamental concepts of object-oriented technologies, UML language elements, software development models, and quality software characteristics.

CO2: *Apply* UML case modeling techniques to analyze system requirements by identifying actors, defining goals, and constructing appropriate use case diagrams.

CO3: *Analyze* flow of events to identify objects, operations, and interactions by constructing sequence and collaboration diagrams.

CO4: *Evaluate* static structural relationships in software design by interpreting class diagrams involving associations, generalizations, aggregations, dependencies, and multiplicities

CO5: *Design* complete UML-based software models by integrating package diagrams, state/activity diagrams, component diagrams, and deployment models to represent logical and physical system architectures.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes											Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1	-	1	2	2	-	-	-	-	1	1	-
CO2	2	3	2	1	2	-	1	-	-	-	-	2	2	-
CO3	3	3	2	2	2	-	-	-	-	-	1	3	3	-
CO4	2	3	2	2	3	1	-	-	-	-	-	2	3	-
CO5	2	2	3	2	3	2	2	3	3	3	2	3	3	3

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

COMPUTER NETWORKS

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Computer Networks	**Course Credit:** 03[2-1-0]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 2^{nd}

➠ Learning Objectives:

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

1. Describe the general principles of data communication.
2. Describe how computer networks are organized with the concept of layered approach.
3. Describe how signals are used to transfer data between nodes.
4. Implement a simple LAN with hubs, bridges and switches.
5. Describe how packets in the Internet are delivered.
6. Analyze the contents in a given data link layer packet, based on the layer concept.
7. Design logical sub-address blocks with a given address block.
8. Decide routing entries given a simple example of network topology
9. Describe what classless addressing scheme is.
10. Describe how routing protocols work.
11. Use C programming language to implement network programs.
12. Design and implement a network protocol.

➠ **Prerequisite:** Analog and Data Communication, Algorithm, and Programming logic.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Introduction to Data Communication	4	11%
Module-II: Physical layer and Media	6	17%
Module-III: Data Link Layer and Medium Access Sub Layer	8	22%
Module-IV: Network Layer	8	22%
Module-V: Transport Layer	6	17%
Module-VI: Application Layer	4	11%

Syllabus Outline:

Module I: Introduction to Data Communication: [4L]

Components, Representation of data and its flow networks, Physical structures, Connection Topology, Protocols and Standards, OSI model, TCP/IP Protocol suite, Addressing.

Module II: Physical layer and Media: [6L]

Analog and Digital data, Signals, Transmission impairment, Data rate limit and Performance, Digital to Digital conversion, Analog to Digital conversion, Digital to Analog conversion, Analog to Analog conversion, Multiplexing and Spectrum Spreading: Multiplexing, Spread Spectrum, Transmission media: Guided Media, Unguided Media, Switching: Introduction, circuit switched networks, packet switched network, switching, structure of a switch.

Module III: Data Link Layer and Medium Access Sub Layer: [8L]

Error Detection and Error Correction - Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols - Stop and Wait, Go back – N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking, Random Access, Multiple access protocols -Pure ALOHA, Slotted ALOHA,CSMA/CD,CDMA/CA. HDLC, Ethernet, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN.

Module IV: Network Layer: [8L]

Logical addressing – IPV4, IPV6; Address mapping – and DHCP–Delivery, Forwarding Unicast Routing protocols: RIP, OSPF, BGP Multicast Routing Protocol:

Module V: Transport Layere: [6L]

Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP)s; Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.

Module VI: Application Layer: [4L]

Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP.

➠ Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books:

Text Books:

1. Computer Networks, Andrew. S. Tanenbaum, 4/e, Prentice Hall of India Private Ltd, 2003.
2. Data Communications and Networking, Behrouz A Forouzan, 4/e, Tata McGraw Hill
3. Education Private Limited.

Reference Books:

1. Data Communications & Networks, Achyut S. Godbole, Tata McGraw Hill Education
2. Private Limited, 2002.
3. Data and Computer Communication, William Stalling, 7/e, Prentice Hall of India Private
4. Ltd, 2007.

➠ Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Understand* fundamental communication principles and network architectures to explain data exchange processes across networked systems.

CO2: *Apply* appropriate encoding, transmission, and switching techniques for ensuring reliable and efficient data communication in diverse networking environments.

CO3: *Evaluate* error control, flow control, and medium access mechanisms to determine their effectiveness in improving link-layer performance.

CO4: *Compare* addressing, routing, and forwarding strategies for selecting suitable network-layer solutions in varied communication scenarios.

CO5: *Design* end-to-end communication solutions by integrating transport and application-layer protocols for supporting quality-driven network services.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	3	-	-	-	-	-	2	-	-	2	-	-
CO2	3	2	2	2	3	-	-	-	-	-	2	-	2	2	-
CO3	3	3	2	2	-	-	-	-	-	2	-	-	2	-	-
CO4	3	2	2	3	-	-	-	-	-	2	-	-	2	-	-
CO5	2	2	3	2	3	-	-	-	-	2	2	-	3	2	2

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

PYTHON PROGRAMMING

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Python Programming	**Course Credit:** 03[2-0-2]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 2^{nd}

➠ Learning Objectives:

1. To provide exposure to basic problem-solving techniques with computers
2. To develop the logical thinking abilities and to propose novel solutions for real world
3. problems through programming language constructs.
4. To deepen the empirical knowledge on applying programming on business domains.

➠ Prerequisite: Basic knowledge of logic building and programming.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Introduction to Python Programming	4	11%
Module-II: Control Structures	6	16%
Module-III: Collections	6	17%
Module-IV: Strings and Regular Expressions	6	17%
Module-V: Functions	8	22%
Module-VI: Handling Exceptions and File Handling	6	17%

➠ Syllabus Outline: .....

Module I: Introduction to Python Programming: [4L]

Introduction to Python, Demo of Interactive and script mode, Tokens in Python – Variables, Keywords, Comments, Literals, Data types, Indentation, Operators and its precedence, Expressions, Input and Print functions. Sequential approach.

Module II: Control Structures: [6L]

Selective statements – if, if-else, nested if, if –elif ladder statements

Iterative statements - while, for, Nested loops, else in loops, break, continue and pass statements.

Module III: Collections: [6L]

List: Create, Access, Slicing, Negative Indices, List Methods, and comprehensions Tuples: Create, Indexing and Slicing, Operations on tuples. Dictionary: Create, add, and replace values, operations on dictionaries. Sets: Create and operations on set.

Module IV: Strings and Regular Expressions: [6L]

Strings: Formatting, Comparison, Slicing, Splitting, Stripping, Negative indices, String functions. Regular expression: Matching the patterns, Search and replace.

Module V: Functions: [8L]

Functions: Types, parameters, arguments: positional arguments, keyword arguments, parameters with default values, functions with arbitrary arguments, Scope of variables: Local and global scope, Recursion and Lambda functions.

Module VI: Handling Exceptions and File Handling: [6L]

Files: Open, Read, Write, Append and Close. Tell and seek methods.

Errors and Exceptions: Syntax Errors, Exceptions, Handling Exceptions, Raising Exceptions, Exception Chaining, User-defined Exceptions, Defining Clean-Up actions

➠ Teaching–Learning Methodology: .....

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books: .....

Text Books:

1. Eric Matthes, Python Crash Course: A Hands-On, Project-Based Introduction to Programming, 2nd Edition, No starch Press, 2019.

Reference Books:

1. Martic C Brown, Python: The Complete Reference, 4th Edition, McGraw Hill Publishers, 2018.
2. Charles Dierbach, Introduction to Computer Science using Python: A Computational Problem-Solving Focus,2nd Edition, Wiley India Edition, 2017.

Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Recall* Python tokens, keywords, data types, operators, and basic syntax.

CO2: *Explain* control structures, decision statements, and core collection types (lists, tuples, sets, dictionaries) and their typical use-cases.

CO3: *Implement* modular Python programs using functions (including recursion and lambdas), collections, and regular expressions to solve practical problems.

CO4: *Analyse* program behaviour by applying debugging, exception-handling, file I/O, and pattern-matching to identify and fix faults and performance bottlenecks.

CO5: *Evaluate* design choices between data structures, algorithms, and I/O strategies and justify suitable alternatives for correctness, efficiency, and maintainability.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	1	1	-	1	-	2	-	1	-	1	2	2	2	-
CO2	3	2	2	1	2	-	2	-	1	-	2	2	3	3	1
CO3	3	2	3	2	3	1	2	2	2	1	2	2	3	3	2
CO4	3	3	2	3	2	1	2	2	2	1	2	2	3	2	2
CO5	3	3	3	2	2	2	2	3	2	2	2	3	3	3	3

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

OPERATING SYSTEMS

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Operating Systems	**Course Credit:** 04[3-1-0]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 2^{nd}

➠ Learning Objectives:

After completing this course, students will be able to understand the mechanisms of the Operating Systems like Process Management, Process Synchronization, Memory Management, File System Implementation, Storage Structures used in OS and Protection Principles. How effectively the OS is utilizing the CPU resources with the help of these mechanisms.

➠ Prerequisite:

Students should have foundational understanding of Computer Organization and Architecture, including CPU operations, memory hierarchy, I/O systems, and interrupts; Basic knowledge of Data Structures and Algorithms for managing process queues, and file structures; Familiarity with at least one programming language.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Introduction	4	8%
Module-II: Process Management	12	25%
Module-III: Process Synchronization and Deadlocks	12	25%
Module-IV: Memory management and Virtual Memory	10	21%
Module-V: File and I/O Systems Management	6	13%
Module-VI: Disk Management	4	8%

Syllabus Outline:

Module I: Introduction: [4L]

Introduction to OS, operating system functions, evaluation of OS, Different types of OS: batch, multi-programmed, time-sharing, real-time, distributed, Network, parallel. System Structure, Computer system operation, I/O structure, storage structure, storage hierarchy, different types of protections, operating system structure (simple, layered, virtual machine), O/S services, system calls.

Module II: Process Management: [12L]

Concept of processes, Process Life Cycle, Process Control Block (PCB), process scheduling: Scheduler, Dispatcher, CPU Scheduling algorithms: scheduling criteria, preemptive & non-preemptive scheduling, scheduling algorithms (FCFS, SJF, SRTF, RR, and priority). Context switch.

Module III: Process Synchronization and Deadlocks: [12L]

Operations on processes: Process Creation, & Termination. Independent and co-operating processes, Inter process communication (IPC): Background, Race condition & Critical Section, Process Synchronization, Techniques to prevent Race Condition: synchronization hardware, classical problems of synchronization, semaphores.

Threads: Threads overview, benefits of threads, user and kernel threads.

System Calls: System Call overview, How system Call Works; functionalities, features, and need of System Calls

Deadlock: deadlock characterization, methods for handling deadlocks, deadlock prevention, deadlock avoidance, deadlock detection, recovery from deadlock.

Module IV: Memory management and Virtual Memory: [10L]

Background, logical vs. physical address space, swapping, contiguous memory allocation, paging, segmentation, segmentation with paging. Virtual Memory background, demand paging, performance, page replacement, page replacement algorithms (FCFS, LRU, Optimal), allocation of frames, thrashing.

Module V: File and I/O Systems Management: [6L]

File concept, access methods, directory structure, file system structure, allocation methods (contiguous, linked, and indexed), and free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency & performance. I/O hardware, polling, interrupts, DMA, application I/O interface (block and character devices, network devices, clocks and timers, blocking and

nonblocking I/O), kernel I/O subsystem (scheduling, buffering, caching, spooling and device reservation, error handling), performance.

Protection & Security Goals of protection, domain of protection, security problem, authentication, one-time password, program threats, system threats, threat monitoring, encryption.

Module VI: Disk Management: [4L]

Disk structure, Seek Time, Rotational Latency, Transfer Time, Disk Access Time, disk scheduling (FCFS, SSTF, SCAN, C-SCAN) , disk reliability, disk formatting, boot block, bad blocks.

➠ Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books:

Text Books:

1. Operating System Principles, Abraham Silberchatz, Peter B.Galvin,Greg Gagne,8th Edition, WileyStudent Edition
2. Operating System-Internals and Design Principles. W. Stallings, 6th Edition, Pearson.

Reference Books:

1. Modern Operating System, Andre w s Tanenbaum, 3rd Edition, PHI
2. Operating System A concept-based Approach, 2nd Edition, D.M.Dhamdhere, TMH.
3. Principle Of Operating Systems, B.LStuart, Cengage Learning, India Edition
4. Operating system, A.s.Godbole, 2nd Edition, TMH.
5. An Introduction to Operating System, P.C.P.bhatt, PHI.

➠ Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Explain* the key functions, structures, and services provided by operating systems.

CO2: *Apply* process and CPU management concepts to demonstrate mechanisms of multitasking

environments.

CO3: *Analyze* synchronization and resource-sharing issues to understand challenges in concurrent system execution.

CO4: *Evaluate* memory and storage management strategies for effective resource utilization.

CO5: *Design* file and I/O management approaches to support reliable and efficient system operations.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	-	-	-	-	-	-	-	2	-	-
CO2	3	3	2	-	1	-	-	-	-	-	-	-	3	2	-
CO3	3	3	3	2	1	-	-	-	-	-	-	-	3	2	-
CO4	3	3	2	2	2	-	-	-	-	-	-	-	3	2	-
CO5	3	3	3	1	2	1	-	1	-	1	-	-	2	3	2

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

MANAGEMENT INFORMATION SYSTEMS

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Management Information Systems	**Course Credit:** 03[3-0-0]
Department: Computer Science Engineering	**Category:** AECC/HSM
Code: XXXXXX	**Semester:** 2^{nd}

➠ Learning Objectives:

1. To describe the role of information technology and decision support systems in business and record the current issues with those of the firm to solve business problems.
2. To introduce the fundamental principles of computer-based information systems analysis and design and develop an understanding of the principles and techniques used.
3. To enable students, understand the various knowledge representation methods and different expert system structures as strategic weapons to counter the threats to business and make business more competitive.
4. To enable the students to use information to assess the impact of the Internet and Internet technology on electronic commerce and electronic business and understand the specific threats and vulnerabilities of computer systems.
5. To provide the theoretical models used in database management systems to answer business questions

➠ Prerequisite: Basic knowledge of mathematics.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Introduction to Management Information Systems	6	21%
Module-II: Process Of Management	6	21%
Module-III: Decision-Making and Information	6	21%
Module-IV: System Analysis and Design	6	21%
Module-V: Development of MIS	6	8%
Module-VI: Decision Support Systems	6	8%

Syllabus Outline:

Module I: Introduction to Management Information Systemse: [6L]

Technology of Information Systems, concepts, definition; role and impact of MIS; role and importance of management; approaches to management; functions of the manager; management as a control system; concepts of data models; database design; client-server architecture.

Module II: Process of Management: [6L]

Planning, organization, staffing, coordination and controlling; management by exception; MIS as a support to management; organization structure and theory; basic model and organization structure; organizational behaviour.

Module III: Decision-Making and Information: [6L]

Decision making concepts, methods, tools and procedures; behavioural concepts in decision-making; organizational decision making; information concepts as a quality product; classification of the information; methods of data and information collection; value of the information; organization and information system concepts, control types; handling system complexity; post implementation problems in systems.

Module IV: System Analysis and Design: [6L]

Need for system analysis; system analysis of existing system; new requirement; system development model; structured system analysis and design; computer system design.

Module V: Development of MIS: [6L]

Development of long-range plans of the MIS; ascertaining the class of the information; determining the information requirement; development and implementation of the MIS; management of quality; MIS factors of success and failure.

Module VI: Decision Support Systems: [6L]

Deterministic systems; artificial intelligence; knowledge-based systems; MIS and the role of DSS; enterprise management systems; enterprise resource planning (ERP); ERP features and benefits; implementation factors of ERP; Internet and Web based information system; Electronic Commerce.

➠ Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books:

Text Books:

1. Management Information Systems, K. C Landon, J. P. Laudon, Prentice Hall, 2000.

Reference Books:

1. Management Information Systems, G. B. Davis, M. H. Olson, McGraw Hill, 1998.

➠ Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Explain* foundational MIS concepts, information system technologies, managerial roles, and organizational structures.

CO2: *Analyze* managerial processes, decision-making models, and information requirements in organizational contexts.

CO3: *Apply* system analysis and design methodologies to evaluate and improve business information systems.

CO4: *Assess* MIS development processes, quality factors, and implementation strategies for effective organizational performance.

CO5: *Evaluate* the role of Decision Support Systems, AI-based systems, and enterprise systems (ERP/E-commerce) for managerial decision-making.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	3	1	1	2	2	1	1	2	1	1
CO2	2	3	2	2	2	2	1	1	2	2	1	1	2	1	1
CO3	2	2	3	2	3	1	1	2	2	1	2	1	3	3	2
CO4	2	2	3	2	2	2	1	2	2	1	2	2	3	3	3
CO5	1	2	2	2	3	2	1	1	2	2	1	3	3	3	3

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

SOFTWARE ENGINEERING USING UML LAB

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Software Engineering using UML Lab	**Course Credit:** 02[0-0-4]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 2^{nd}

➠ List of Practicals: .....

Draw standard UML diagrams using an UML modelling tool for a given case study and map design to code and implement a 3 layered architecture. Test the developed code and validate whether the SRS is satisfied

1. Identify a software system that needs to be developed.
2. Document the Software Requirements Specification (SRS) for the identified system.
3. Identify use cases and develop the Use Case model.
4. Identify the conceptual classes and develop Class Diagram
5. Using the identified scenarios, find the interaction between objects and represent them using UML Sequence Diagrams
6. Draw relevant State Chart and Activity Diagrams for the same system.
7. Implement the system as per the detailed design
8. Test the software system for all the scenarios identified as per the usecase diagram

PYTHON PROGRAMMING LAB

Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Python Programming Lab	**Course Credit:** 02[0-0-4]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 2^{nd}

List of Practicals: .....

1. Sequential programs with python tokens, operators and expressions
2. Selectional and Looping constructs
3. List,Tuples, Dictionary and Sets
4. String Manipulation and Regular Expression
5. Functions, Recursion and Lamda functions
6. Files
7. Exception Handling

OPERATING SYSTEMS LAB

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Operating Systems Lab	**Course Credit:** 02[0-0-4]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 2^{nd}

➠ List of Practicals: .....

Section 1:

1. Write a shell script to take the name of the user as input and print it.
2. Write a shell script to multiply two numbers and display the output.
3. Write a shell script program to emulate the calculator function.
4. Write a shell script that will find the maximum from the given three no.
5. Write a shell script that will find the GCD of two given numbers.
6. Write a shell script to generate a Fibonacci series of length with the first two no. of the series is 3 and 5 respectively.
7. Write a Shell script to take 'n' number of elements in an array and print the third largest number. Value of 'n' must be taken from the terminal.
8. Store 'n' number of elements in an array and find out the sum of the array elements. Value of 'n' must be taken from the terminal.
9. Write a shell program that will accept 10 numbers from the
10. terminals and will search the position of a given no in the supplied nos.
11. Write a program in C under Linux to create a file.
12. Write a shell script program to search an integer in an array using linear search.

Section 2:

1. Write a C Program that will create a child process. Then print the process id & parent process id both from the child as well as from the parent.
2. Write a C program that will create a child process. Then modify the value of a globally defined variable from the child process and print the value of the variable from the parent process.
3. Write a c program that will take the name of person as command line argument and then it will print hallo name. Then write another program that will create a process using fork(). Then execute

the previously created c program (hallo program) by the child process.

4. Write a program in C under Linux to copy the content of one file to another from command line.
5. Write a program in C to implement LRU page replacement algorithm
6. Write a program in C to implement CPU scheduling using Round Robin Scheduling algorithm
7. Write a program in C to implement CPU scheduling using FCFS Scheduling algorithm
8. Write a program in C to implement CPU scheduling using SJF Scheduling algorithm.

Section 3:

1. Write a C program for implementing the Producer Consumer problem using Thread Synchronization.
2. Write a C program to count a number from 1 to 20 using two threads (Thread 1 and Thread 2) where the prime numbers are printed by Thread 1 and non prime numbers are printed by Thread 2.

Section 4:

1. Write a program in C that demonstrates how two processes can share a variable using semaphore.
2. Write a C program to implement Semaphore to print from a parent as well as a child process, where both parent and child will print two consecutive words from a sentence.
3. Write a program in C to solve the Producer Consumer problem using POSIX semaphore.

Section 5:

1. Write Unix Commands to do the following directory manipulation.
   a) Display the absolute path of your home directory.
   b) Create a new subdirectory called ‘Sister Nivedita University’ in your home directory.
   c) Create a new subdirectory called ‘Student’ in Sister Nivedita University.
   d) Create a new subdirectory called ‘Teacher’ in Sister Nivedita University.
   e) Display the contents of the directory ‘Sister Nivedita University’.
   f) Delete the directory ‘Teacher’.
   g) Display the contents of the directory ‘Sister Nivedita University’ in detail .
2. Write a program to create a pipe between parent and child and to send data down the pipe.
3. Write a program to convert lower case to upper case using FIFO pipe where the client sends a string in lower case to the server and the server responds with the string in Upper case back to the client.
4. Write a program to implement IPC using shared memory between two processes.
5. Write a program to implement IPC using message queue between two processes.
6. Write a C program to calculate the seek time by applying FCFS, SSTF, SCAN,C-SCAN algorithms

SECOND YEAR

SEMESTER-III

Sl. No.	Paper Name	Code	Category	Credit	Type			
					L	T	P	S
1	Artificial Intelligence		CC/PCC	3	2	1	0	0
2	Theory of Computation		CC/PCC	3	2	1	0	0
3	Elective		DSE/PEC	3	0	0	0	0
4	Operation Research		BS	2	2	0	0	0
5	Accounting and Management Control		AECC/HSM	3	2	1	0	0
6	Customer Relationship Management using Salesforce		AECC/HSM	2	1	0	2	0
7	Foreign Language – III (German/ Spanish/ Japanese)		USC/MUS	2	2	0	0	0
8	Minor Project-I		SEC/PSE	6	0	0	0	12
Practical								
9	Artificial Intelligence Lab		CC/PCC	1	0	0	2	0
				Total Credit=25				

Elective

A. Data Warehousing and Data Mining
B. Compiler Design
C. Distributed Database System
D. AI and Neural Network
E. Cryptography and Network Security
F. Machine Learning
G. Internet of Things
H. Cloud Computing

ARTIFICIAL INTELLIGENCE

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Artificial Intelligence	**Course Credit:** 03[3-0-0]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 3^{rd}

➠ Learning Objectives:

On completion of the course, student will be able to: Develop problem-solving ability, incorporate knowledge representation, allow continuous learning, encourage social Intelligence, achieve general intelligence, Promote synergy between humans and AI

➠ Prerequisite:

Basic computer knowledge and Data Structure and Algorithm.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Introduction	4	12%
Module-II: Search Techniques	6	18%
Module-III: Knowledge & Reasoning	6	12%
Module-IV: Probabilistic Reasoning	6	18%
Module-V: Natural Language Processing	6	20%
Module-VI: Expert Systems	8	20%

➠ Syllabus Outline: .....

Module I: Introduction: [4L]

Problems of AI, AI technique, Tic- Tac - Toe problem, games and game playing approaches. Agents & environment, nature of environment, structure of learning agents. Problem space, state space search, problem characteristics, issues in the design of search programs.

Module II: Search Techniques: [6L]

Solving problems by searching: problem solving agents, searching for solutions; uniform search strategies: breadth first search, depth first search, depth limited search, bidirectional search, comparing uniform search strategies. Greedy best-first search, A $*$ search, memory bounded heuristic search: local search algorithms & optimization problems: Hill climbing search, simulated annealing search, local beam search, local search for constraint satisfaction problems. Adversarial Search: Games, optimal decisions & strategies in games, the minimax search procedure, alpha-beta pruning, additional refinements iterative deepening.

Module III: Knowledge & Reasoning: [6L]

Knowledge representation issues, representation & mapping, approaches to knowledge representation, issues in knowledge representation. Using Predicate Logic: Representing simple fact in logic, representing instant & ISA relationship, computable functions & predicates, resolution, natural deduction.

Module IV: Probabilistic Reasoning: [6L]

Representing knowledge in an uncertain domain, the semantics of Bayesian networks, Dempster-Shafer theory, Fuzzy sets & fuzzy logics.

Module V: Natural Language Processing: [6L]

Introduction, syntactic processing, semantic analysis, discourse & pragmatic processing.

Module VI: Expert Systems: [8L]

Representing and using domain knowledge, expert system shells, knowledge acquisition. Learning: Forms of learning, inductive learning, learning decision trees, explanation-based learning, learning using relevance information, neural net learning & genetic learning.

➠ Teaching–Learning Methodology: .....

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books: .....

1. Russell, Stuart, and Peter Norvig. Artificial Intelligence: A Modern Approach. 3rd ed., Pearson, 2016.
2. Bishop, Christopher M. Pattern Recognition and Machine Learning. Springer, 2006.

3. Goodfellow, Ian, et al. Deep Learning. MIT Press, 2016.

Reference Books:

1. Sutton, Richard S., and Andrew G. Barto. Reinforcement Learning: An Introduction. 2nd ed., MIT Press, 2018.
2. Poole, David, and Alan Mackworth. Artificial Intelligence: Foundations of Computational Agents. Cambridge University Press, 2017.
3. Murphy, Kevin P. Machine Learning: A Probabilistic Perspective. MIT Press, 2012.
4. Heaton, Jeff. Artificial Intelligence for Humans. Create Space Independent Publishing Platform, 2015.

Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Define* fundamental concepts, problem-solving approaches, agents, and environments in Artificial Intelligence.

CO2: *Apply* appropriate search strategies, game-playing techniques for solving AI-based problems.

CO3: *Analyze* knowledge representation methods, logical reasoning, probabilistic reasoning techniques for various AI applications.

CO4: *Evaluate* natural language processing methods, expert systems, learning algorithms for their effectiveness in real-world problem domains.

CO5: *Design* AI-based systems with suitable representation, search, reasoning, learning techniques.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	1	2	-	-	-	-	-	2	1	-
CO2	2	3	2	2	2	-	2	-	1	-	1	-	3	2	1
CO3	2	3	2	3	2	-	2	-	1	1	1	-	3	2	1
CO4	2	2	3	2	3	1	2	1	2	2	2	-	2	3	2
CO5	2	3	3	2	3	1	2	2	2	2	3	-	3	3	3

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

THEORY OF COMPUTATION

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Theory of Computation	**Course Credit:** 03[3-0-0]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 3^{rd}

➠ Learning Objectives: After completing this course, students will be able to:

1. Understand the foundational concepts of automata theory, formal languages, and computability, including their role in computer science.
2. Know different computational models such as finite automata, pushdown automata, and Turing machines, and relate them to corresponding language classes.
3. Construct automata and grammars for regular, context-free, and recursively enumerable languages and apply transformation and simplification techniques.
4. Examine decidability and undecidability problems, including reductions, diagonalization, and the halting problem.

➠ Prerequisite:

Students should have solid understanding of Discrete Mathematics, including logic, sets, relations, functions, graphs, and proof techniques (induction, contradiction); Basic knowledge of Data Structures and Algorithms, particularly strings, recursion, and computational problem-solving.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Finite Automata	10	27%
Module-II: Regular Languages and Regular Grammars	6	17%
Module-III: Context-Free Languages	6	17%
Module-IV: Linear Bounded Automata and Turing Machine	6	17%
Module-V: Recursive and Recursively Enumerable language	4	11%
Module-VI: Computational Complexity	4	11%

Syllabus Outline:

Module I: Finite Automata: [10L]

Introduction, Finite state machine (FSM), state table & state assignments, Finite Automata, Types of Finite Automata: DFA, NFA, $\varepsilon-$NFA; DFA: mathematical representation, transition diagram & table, Construction of DFA, DFA Minimization, complement of DFA. NFA: mathematical representation, transition diagram & table, Construction of NFA, Conversion of NFA to DFA, complement of NFA; $\varepsilon-$NFA: mathematical representation, Construction of $\varepsilon-$NFA, Conversion of $\varepsilon-$NFA to NFA.

Moore Machine and Mealy Machine: Mathematical, diagramatical and tabular Representation of Moore Machine and Mealy Machine, Construction of more Machine and Mealy machine, Transformation of a Moore M/C to Mealy M/C and vice versa.

Module II: Regular Languages and Regular Grammars: [6L]

Classification of grammar, Regular Expressions, Regular Grammars, Regular Expressions to Finite Automata, Finite Automata to Regular Expressions, Properties of Regular Languages, Identifying regular and Non-regular Languages.

Module III: Context-Free Languages: [6L]

Definition of Context-Free Grammars (CFG), Examples of Context-Free Languages Leftmost and Rightmost Derivations, Derivation Trees, Relation Between Sentential Forms and Derivation Trees,, Ambiguity in Grammars and Languages, CFG Simplification, Chomsky Normal Form (CNF), conversion of CFG to CNF, Greibach Normal Form (GNF), Pushdown Automata (PDA): Definition, Language Accepted by PDA, Pumping Lemma.

Module IV: Linear Bounded Automata and Turing Machine: [6L]

Linear Bounded Automata: Concept of Context-sensitive, Closure properties of Context-sensitive Language,

Introduction and basic concepts, Representation of Turing Machine, Design of Turing Machine, Linear bounded automata, and languages.

Module V: Recursive and Recursively Enumerable language: [4L]

Recursive Language (REC), Recursively Enumerable Language (RE), Turing Machine Halting Problem, Decidable and undecidable languages: Decidable languages, Undecidabile languages.

Module VI: Computational Complexity [4L]

Types of Complexity Classes: P, NP, NP hard and NP complete; Features and examples of P, NP, NP hard and NP complete.

➠ Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books:

Text Books:

1. Hopcroft and Ullman, "Introduction to Automata Theory, Languages and Computation", 2nd edition, Pearson/Prentice Hall India, 2007.

Reference Books:

1. K.L.P. Mishra and N.Chandrasekaran, "Theory of Computer Science: Automata, Languages and Computation", 2nd edition, Pearson/Prentice Hall India, 2004.
2. Martin J. C., "Introduction to Languages and Theory of Computations", 2nd edition, Tata McGraw Hill, 2005.

➠ Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Define* formal language theory and foundational models of computation.

CO2: *Design* various computational models such as finite automata, grammars, and Turing machines for solving computational problems.

CO3: *Implement* formal techniques for determining the regularity, context-freeness, or decidability of languages and problems.

CO4: *Analyze* the computational power and limitations of different automata and machine models with respect to language recognition and problem-solving capabilities.

CO5: *Classify* computational problems based on complexity classes such as P, NP, NP-hard, and NP-complete.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	1	-	1	1	-	-	1	-	-	2	-	-
CO2	2	3	3	1	2	-	-	-	-	-	1	1	3	2	1
CO3	2	3	2	2	3	-	1	-	-	-	-	-	3	2	-
CO4	3	3	1	2	1	1	1	-	-	1	-	-	3	1	-
CO5	3	3	1	2	1	1	1	-	-	1	-	-	3	1	-

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

ELECTIVE (A): DATA WAREHOUSING AND DATA MINING

Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Data Warehousing and Data Mining	**Course Credit:** 03[3-0-0]
Department: Computer Science Engineering	**Category:** DSE/PEC
Code: XXXXXX	**Semester:** 3^{rd}

Learning Objectives: After completing this course, students will be able to:

1. Be familiar with the mathematical foundations of data mining tools.
2. Understand and implement classical models and algorithms in data warehouses and data mining Characterize the kinds of patterns that can be discovered by association rule mining, classification, and clustering.
3. Master data mining techniques in various applications and develop skills in selecting the appropriate data mining algorithm for solving practical problems.

Prerequisite: Students should have basic concept of mathematics.

Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Data warehousing, business analysis and on-line analytical processing (OLAP)	4	11 %
Module-II: Data mining introduction	6	17%
Module-III: Data mining frequent pattern analysis	6	17%
Module-IV: Classification	8	22%
Module-V: Clustering	8	22%
Module-VI: WEKA tool	4	11%

Syllabus Outline: .....

Module I: Data warehousing, business analysis and on-line analytical processing (OLAP): [4L]

Basic Concepts-Data Warehousing Components Building a Data Warehouse, Database Architectures for Parallel Processing Parallel DBMS Vendors Multidimensional Data Model-Data Warehouse Schemas for Decision Support, Concept Hierarchies -Characteristics of OLAP Systems, Typical OLAP Operations, OLAP and OLTP.

Module II: Data mining introduction: [6L]

Introduction to Data Mining Systems Knowledge Discovery Process Data Mining Techniques Issues applications- Data Objects and attribute types, Statistical description of data, Data Preprocessing Cleaning, Integration, Reduction, Transformation and discretization, Data Visualization, Data similarity and dissimilarity measures.

Module III: Data mining frequent pattern analysis: [6L]

Mining Frequent Patterns, Associations and Correlations Mining Methods- Pattern Evaluation Method, Pattern Mining in Multilevel, Multi-Dimensional Space Constraint Based Frequent Pattern Mining, Classification using Frequent Patterns

Module IV: Classification: [8L]

Decision Tree Induction, Bayesian Classification, Rule Based Classification. Classification by Back Propagation Support Vector Machines Lazy Learners, Model Evaluation and Selection-Techniques to improve Classification Accuracy.

Module V: Clustering: [8L]

Clustering Techniques, Cluster analysis-Partitioning Methods, Hierarchical Methods, Density Based Methods, Grid Based Methods, Evaluation of clustering, Clustering high dimensional data- Clustering with constraints, Outlier analysis-outlier detection methods.

Module VI: WEKA tool: [4L]

Datasets Introduction, Iris plants database, Breast cancer database, Auto imports database Introduction to WEKA, The Explorer Getting started, Exploring the explorer, Learning algorithms, Clustering algorithms, Association rule learners.

Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books:

Text Books:

1. Jiawei Han and Micheline Kamber, Mining Concepts and Techniques, Third Edition, Elsevier, 2012.
2. Alex Berson and Stephen J.Smith, ta Warehousing, Data Mining & Tata McGraw Hill Edition, 35th Reprint 2016.

Reference Books:

1. K.P. Soman, Shyam Diwakar and V. Ajay, into Data Mining Theory and Practice, Eastern Economy Edition, Prentice Hall of India, 2006.
2. Ian H.Witten and Eibe Frank, ata mining: Practical Machine Learning Tools and Techniques, Elsevier, Second Edition.

➠ Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Explain* the fundamental concepts of data warehousing, OLAP operations, and multidimensional data models for analytical processing.

CO2: *Apply* data preprocessing, cleaning, integration, transformation, and visualization techniques for preparing datasets for mining.

CO3: *Analyze* frequent pattern mining approaches including association rule mining, correlation analysis, and constraint-based mining in multidimensional spaces.

CO4: *Evaluate* various classification algorithms such as decision trees, Bayesian methods, SVM, and rule-based classification for model performance improvement.

CO5: *Implement* and *compare* clustering techniques, outlier detection methods, and practical data mining algorithms using WEKA for real-world datasets.

➠ CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	1	1	1	2	1	1	1	2	1	1
CO2	2	3	1	2	3	1	1	1	2	1	1	1	3	2	1
CO3	2	3	2	2	3	1	1	1	2	1	2	2	3	3	2
CO4	2	3	3	3	3	1	1	2	2	1	2	3	3	3	3
CO5	2	2	3	2	3	1	1	2	2	1	2	3	3	3	3

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

ELECTIVE (B): COMPILER DESIGN

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Compiler Design	**Course Credit:** 03[3-0-0]
Department: Computer Science Engineering	**Category:** DSE/PEC
Code: XXXXXX	**Semester:** 3^{rd}

➠ Learning Objectives: On completion of the course, student will be able to:

1. Understand the structure of a compiler, and how the source and target languages influence various choices in its design.
2. Understand A new appreciation for programming language features and the implementation challenges they pose, as well as for the actual hardware architecture and the run-time system in which your generated code executes.
3. Understand some specific components of compiler technology, such as lexical analysis, grammars and parsing, type-checking, intermediate representations, static analysis, common optimizations, instruction selection, register allocation, code generation, and run-time organization.

➠ Prerequisite:

Before learning the concepts of Compiler Design, you should have a basic knowledge Programming for problem solving and Formal Languages and Automata Theory etc.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Introduction to Compiling	2	5%
Module-II: Lexical Analysis	6	17%
Module-III: Syntax Analysis	8	22%
Module-IV: Syntax directed translation and Type	8	22%
Module-V: Run time environments and Intermediate Code Generation	6	17%
Module-VI: Code optimization and Code generations	6	17%

Syllabus Outline:

Module I: Introduction to Compiling: [2L]

Compilers, Analysis of the source program, the phases of the compiler, Cousins of the compiler.

Module II: Lexical Analysis: [6L]

The role of the lexical analyzer, Tokens, Patterns, Lexemes, Input buffering, Specifications of a token, Recognition of a tokens, Finite automata, from a regular expression to an NFA, From a regular expression to NFA, From a regular expression to DFA, Design of a lexical analyzer generator (Lex).

Module III: Syntax Analysis: [8L]

The role of a parser, Context free grammars, writing a grammar, Top down Parsing, Non recursive Predictive parsing (LL), Bottom up parsing, Handles, Viable prefixes, Operator precedence parsing, LR parsers (SLR, LALR), Parser generators (YACC). Error Recovery strategies for different parsing techniques.

Module IV: Syntax directed translation and Type: [8L]

Syntax director definitions, Construction of syntax trees, Bottom-up evaluation of S attributed definitions, L attributed definitions, Bottom-up evaluation of inherited attributes. Type systems, Specification of a simple type checker, Equivalence of type expressions, Type conversions.

Module V: Run time environments and Intermediate Code Generation: [6L]

Source language issues (Activation trees, Control stack, scope of declaration, Binding of names), Storage organization (Subdivision of run-time memory, Activation records), Storage allocation strategies, Parameter passing (call by value, call by reference, copy restore, call by name), Symbol tables, dynamic storage allocation techniques. Intermediate languages, Graphical representation, Three-address code, Implementation of three address statements (Quadruples, Triples, Indirect triples).

Module VI: Code optimization and Code generations: [6L]

Introduction, Basic blocks & flow graphs, Transformation of basic blocks, Dag representation of basic blocks, the principal sources of optimization, Loops in flow graph, Peephole optimization. Issues in the design of code generator, a simple code generator, Register allocation & assignment.

Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)

- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books:

Text Books:

1. Micheal McTear, Conversational AI: Dialogue Systems, Conversational Agents and chatbots, 2020, 1st Edition, Morgan and Claypool.
2. Luis Fernando D Haro, Zoraida Callejas, Satosh Nakamura, Conversational Dialogue Systems for the Next Decade, 2021,1st Edition, Springer.

Reference Books:

1. Srini Janarthanam, Chatbots and Conversational UI Development, 2017, 1st Edition, Packt Publishers.
2. Diana Perez-marin and Ismael Pascual-Nieto, Conversational Agents And Natural Language Interaction, 2011, 1st Edition, IGI Global publishers

➠ Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Understand* the fundamental concepts, structure, and functioning of a compiler, including its major phases and their roles in program translation.

CO2: *Apply* formal language principles, automata theory, and grammar-based techniques to perform lexical, syntactic, and semantic analysis of source programs.

CO3: *Analyze* translation mechanisms, attribute evaluation strategies, type systems, and runtime environments to ensure correct and efficient program execution.

CO4: *Design* intermediate representations, symbol table structures, and code generation strategies required for transforming high-level programs into executable code.

CO5: *Evaluate* and optimize generated code using standard optimization techniques to improve performance, memory usage, and computational efficiency.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	2	1	-	-	-	-	-	2	1	2	1	-
CO2	3	3	2	3	-	-	-	-	-	-	2	2	3	2	-
CO3	2	3	2	3	-	-	-	-	-	-	3	2	3	2	-
CO4	2	2	3	3	-	-	1	1	1	1	2	3	3	3	1
CO5	1	2	2	3	-	-	1	-	1	-	3	3	3	3	1

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

ELECTIVE (C): DISTRIBUTED DATABASE SYSTEM

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Distributed Database System	**Course Credit:** 03[3-0-0]
Department: Computer Science Engineering	**Category:** DSE/PEC
Code: XXXXXX	**Semester:** 3^{rd}

➠ Learning Objectives: After completing this course, students will be able to:

1. Enrich the previous knowledge of database systems and expose the need for distributed database technology to confront the deficiencies of centralized database systems.
2. Introduce basic principles and implementation techniques of distributed database systems.
3. Equip students with principles and knowledge of parallel and object-oriented databases.
4. Understand query processing and optimization; distributed transaction management and reliability; parallel and object database management systems.

➠ Prerequisite:

The primary prerequisite for this course is basic concept of mathematics.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Introduction	6	17%
Module-II: Logical Time	2	5%
Module-III: Message Ordering & Snapshots	8	22%
Module-IV: Distributed Mutex & Deadlock	8	22%
Module-V: Recovery & Consensus	6	17%
Module-VI: P2P & Distributed Shared Memory	6	17%

Syllabus Outline:

Module I: Introduction: [6L]

Definition, Relation to computer system components Motivation Relation to parallel systems, Message-passing systems versus sharedmemorysystems, Primitives for distributed communication Synchronous versus asynchronous executions Design issues and challenges. A model of distributed computations: A distributed program A model of distributed executions Models of communication networks Global stateCuts Past and future cones of an event Models of process communications.

Module II: Logical Time: [2L]

A framework for a system of logical clocks Scalar time Vectortime Physical clock synchronization: NTP.

Module III: Message Ordering & Snapshots: [8L]

Message ordering and group communication: Message ordering paradigms Asynchronous execution with synchronous communication Synchronous program order on an asynchronous system Group communication Causal order (CO) Total order. Global state and snapshot recording algorithms: Introduction Systemmodel and definitions Snapshot algorithms for FIFO channels

Module IV: Distributed Mutex & Deadlock: [8L]

Distributed mutual exclusion algorithms:Introduction Preliminaries algorithm Ricart-Agrawala algorithm, algorithm Suzuki broadcast algorithm. Deadlock detection in distributed systems: Introduction System model Preliminaries Models of deadlocks classification Algorithms for the single resource model, the AND model, and the OR model.

Module V: Recovery & Consensus: [6L]

Check-pointing and rollback recovery: Introduction Background and definitions, Issues in failure recovery Checkpoint-based recovery, Log-based rollback recovery, coordinated check-pointing algorithm, Algorithm for asynchronous check-pointing and recovery. Consensus and agreement algorithms: Problem definition, Overview of results, Agreement in a failure free system Agreement in synchronous, systems with failures.

Module VI: P2P & Distributed Shared Memory: [6L]

Peer-to-peer computing and overlay graphs: Introduction Data indexing and overlays Chord Content addressable networks Tapestry. Distributed shared memory: Abstraction and advantages Memory consistency models Shared memory Mutual Exclusion.

Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

Text & Reference Books:

Text Books:

1. Pradeep Operating Systems: Concepts and Prentice Hall of India, 2007.
2. Mukesh Singhal and Niranjan G. Shivaratri. Advanced concepts in operating systems. McGraw-Hill, Inc., 1994.
3. Tanenbaum A.S., Van Steen M., Systems: Principles and Pearson Education, 2007.

Reference Books:

1. Liu M.L.,Computing, Principles, and Pearson Education, 2004.
2. Nancy A Lynch, Morgan Kaufman Publishers, USA, 2003.

Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Understand* fundamental concepts of distributed database systems and architectures

CO2: *Analyze* distributed query processing, transaction management, and concurrency control

CO3: *Apply* distributed database design, replication, and fragmentation techniques

CO4: *Evaluate* performance, reliability, and security of distributed databases

CO5: *Demonstrate* practical implementation of distributed database systems using contemporary software

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	-	-	-	-	-	-	-	-	3	2	1
CO2	2	3	2	2	1	1	-	1	1	-	-	-	3	2	2
CO3	3	2	3	2	2	2	1	2	1	-	-	-	3	3	2
CO4	2	3	2	2	3	2	-	1	2	-	-	-	2	3	2
CO5	3	2	3	2	2	1	1	2	1	-	-	-	3	2	3

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

ELECTIVE (D): AI AND NEURAL NETWORK

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: AI and Neural Network	**Course Credit:** 03[3-0-0]
Department: Computer Science Engineering	**Category:** DSE/PEC
Code: XXXXXX	**Semester:** 3^{rd}

➠ Learning Objectives: After completing this course, students will be able to:

1. Understand the context of neural networks and deep learning
2. Know how to use a neural network
3. Understand the data needs of deep learning
4. Have a working knowledge of neural networks and deep learning Explore the parameters for neural networks

➠ Prerequisite: Students should have basic computer knowledge and Data Structure and Algorithm.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Introduction	2	5%
Module-II: Searching	10	28%
Module-III: Knowledge Representation	4	11%
Module-IV: Characteristics of Neural Networks, Historical Development of Neural Networks Principles, Artificial Neural Networks	6	17%
Module-V: Feedforward Neural Networks	10	28%
Module-VI: Competitive Learning Neural Networks & Complex pattern Recognition	4	11%

Syllabus Outline:

Module I: Introduction: [2L]

AI problems, the foundation of AI and history of AI intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, structure of agents, problem-solving agents, problem formulation.

Module II: Searching: [10L]

Searching for solutions, uniformed search strategies Breadth-first search, depth-first Search. Search with partial information (Heuristic search) Greedy best-first search, $A*$ search Game Playing: Adversarial search, Games, minimax, algorithm, optimal decisions in multiplayer games, Alpha-Beta pruning, Evaluation functions, cutting of search.

Module III: Knowledge Representation: [4L]

Reasons logical Agents, Knowledge Based Agents, the Wumpus world, logic, propositional logic, Resolution patterns in propositional logic, Resolution, Forward & Backward. Chaining. First-order logic. Inference in first-order logic, propositional Vs. first-order inference, unification & lifts forward chaining, Backward chaining, Resolution.

Module IV: Characteristics of Neural Networks, Historical Development of Neural Networks Principles, Artificial Neural Networks: [6L]

Terminology, Models of Neuron, Topology, Basic Learning Laws, Pattern Recognition Problem, Basic Functional Units, Pattern Recognition Tasks by the Functional Units.

Module V: Feedforward Neural Networks: [10L]

Introduction, Analysis of pattern Association Networks, Analysis of Pattern Classification Networks, Analysis of Pattern Storage Networks. Analysis of Pattern Mapping Networks. Feedback Neural Networks: Introduction, Analysis of Linear Auto associative FF Networks, Analysis of Pattern Storage Networks.

Module VI: Competitive Learning Neural Networks & Complex pattern Recognition: [4L]

Introduction, Analysis of Pattern Clustering Networks, Analysis of Feature Mapping Networks, and Associative Memory.

Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

Text & Reference Books:

Text Books:

1. Artificial Intelligence:A Modern Approach. Second Edition, Stuart Russel, Peter Norvig, PHI/ Pearson Education.
2. Artificial Intelligence, 2nd Edition, E.Rich, and K.Knight (TMH).
3. Artificial Intelligence and Expert Systems Patterson PHI.

Reference Books:

1. Neural Networks Simon Haykin PHI.
2. Artificial Intelligence, 3rd Edition, Patrick Henry Winston., Pearson Edition.

Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Explain* foundational concepts of Artificial Intelligence, intelligent agents, environments, problem formulation, and classical search strategies.

CO2: *Apply* uninformed, heuristic, and adversarial search techniques (BFS, DFS, A*, Minimax, Alpha–Beta pruning) to solve computational and game-based problems.

CO3: *Analyze* logical and knowledge-based reasoning systems using propositional logic, first-order logic, and inference mechanisms for real-world decision-making.

CO4: *Evaluate* neural network architectures, neuron models, learning laws, and pattern recognition mechanisms to determine suitability for specific AI tasks.

CO5: *Implement* feedforward, feedback, and competitive-learning neural network models for clustering, classification, feature mapping, and associative memory applications.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	1	1	-	-	-	1	-	-	1	2	1	-
CO2	2	3	1	2	1	-	-	-	1	-	-	1	2	1	-
CO3	2	3	2	3	1	-	-	-	1	-	-	1	2	1	-
CO4	3	2	2	1	3	-	-	-	1	-	-	2	3	2	1
CO5	2	2	3	2	2	-	-	-	1	-	-	3	3	2	1

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

ELECTIVE (E): CRYPTOGRAPHY AND NETWORK SECURITY

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Cryptography and Network Security	**Course Credit:** 03[3-0-0]
Department: Computer Science Engineering	**Category:** DSE/PEC
Code: XXXXXX	**Semester:** 3^{rd}

➠ Learning Objectives: After completing this course, students will be able to:

1. To understand basics of Cryptography and Network Security.
2. To be able to secure a message over insecure channel by various means.
3. To learn about how to maintain the Confidentiality, Integrity and Availability of a data.
4. To understand various protocols for network security to protect against the threats in the networks.

➠ Prerequisite:

Understanding of mathematical principles, such as linear algebra, number theory, and combinatorics.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Attacks on Computers & Computer Security	4	11%
Module-II: Cryptography: Concepts & Techniques	4	11%
Module-III: Symmetric Key Algorithm	6	17%
Module-IV: Asymmetric Key Algorithm, Digital Signature and RSA	10	27%
Module-V: Internet Security Protocols, User Authentication	6	17%
Module-VI: Electronic Mail Security and Firewall	6	17%

Syllabus Outline:

Module I: Attacks on Computers & Computer Security: [4L]

Introduction, Need for Security, Security approaches, Principles of Security, Types of attack

Module II: Cryptography: Concepts & Techniques: [4L]

Encryption & Decryption, Symmetric & Asymmetric key Cryptography, Key Range & Key Size Introduction, Plaintext & Cipher text, Substitution Techniques, Transposition Techniques.

Module III: Symmetric Key Algorithm: [6L]

Introduction, Algorithm types & Modes, Overview of Symmetric Key Cryptography, DES(Data Encryption Standard) algorithm, IDEA(International Data Encryption Algorithm) algorithm, RC5(Rivest Cipher 5) Algorithm.

Module IV: Asymmetric Key Algorithm, Digital Signature and RSA: [10L]

Introduction, Overview of Asymmetric key Cryptography, RSA algorithm, Symmetric & Asymmetric key Cryptography together, Digital Signature, Basic concepts of Message Digest and Hash Function (Algorithms on Message Digest and Hash function not required)

Module V: Internet Security Protocols, User Authentication: [6L]

Basic Concepts, SSL protocol, Authentication Basics, Password, Authentication Token, Certificate based Authentication, Biometric Authentication.

Module VI: Electronic Mail Security and Firewall: [6L]

Basics of mail security, Pretty Good Privacy, S/MIME, Introduction to Firewall, Types of firewall, Firewall Configurations, DMZ Network

Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

Text & Reference Books:

Text Books:

1. Stallings, William. Cryptography and Network Security: Principles and Practice. 7th ed., Pearson,

2017.

2. Paar, Christof, and Jan Pelzl. Understanding Cryptography: A Textbook for Students and Practitioners. 2nd ed., Springer, 2010.
3. Kaufman, Charlie, Radia Perlman, and Mike Speciner. Network Security: Private Communication in a Public World. 2nd ed., Prentice Hall, 2002.
4. Schneier, Bruce. Applied Cryptography: Protocols, Algorithms, and Source Code in C. 2nd ed., Wiley, 1996.

Reference Books:

1. Ferguson, Niels, Bruce Schneier, and Tadayoshi Kohno. Cryptography Engineering: Design Principles and Practical Applications. Wiley, 2010.
2. Forouzan, Behrouz A., and Debdeep Mukhopadhyay. Cryptography and Network Security. McGraw-Hill Education, 2018.

➠ Course Outcome (CO): .....

Upon successful completion of this course, students will be able to:

CO1: *Explain* the fundamental security concepts, principles of computer security, and various types of cyberattacks.

CO2: *Apply* classical and modern cryptographic techniques including substitution, transposition, symmetric-key, and asymmetric-key algorithms.

CO3: *Analyze* the working of encryption algorithms such as DES, IDEA, RC5, RSA, and digital signatures to determine their strengths and limitations.

CO4: *Evaluate* Internet security protocols, authentication mechanisms, and message-digest techniques for secure communication.

CO5: *Implement* basic email security mechanisms and firewall configurations, and assess their effectiveness in real-world network environments.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	3	2	-	1	2	-	1	2	1	-
CO2	3	3	2	2	3	2	1	-	1	1	-	1	3	3	1
CO3	3	3	2	3	2	2	1	-	1	1	-	2	3	3	2
CO4	2	3	3	3	3	2	1	-	2	2	-	2	3	3	2
CO5	2	2	3	2	3	2	1	1	2	2	1	3	3	3	3

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

ELECTIVE (F): MACHINE LEARNING

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Machine Learning	**Course Credit:** 03[3-0-0]
Department: Computer Science Engineering	**Category:** DSE/PEC
Code: XXXXXX	**Semester:** 3^{rd}

➠ Learning Objectives: After completing this course, students will be able to:

1. Understand the basic theory underlying machine learning.
2. Formulate machine learning problems corresponding to different applications.
3. Understand a range of machine learning algorithms along with their strengths and weaknesses.
4. Apply machine learning algorithms to solve problems of moderate complexity.
5. Apply the algorithms to a real-world problem, optimize the models learned and report on the expected accuracy that can be achieved by applying the models.

➠ Prerequisite: Knowledge of Artificial Intelligence, Linear algebra, Mathematical logic

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Introduction to Machine Learning	2	6%
Module-II: Feature Engineering	4	11%
Module-III: Classification	10	28%
Module-IV: Clustering	12	32%
Module-V: Machine Learning System Design	6	17%
Module-VI: Case studies	2	6%

Syllabus Outline:

Module I: Introduction to Machine Learning: [2L]

Basic Concepts of Machine Learning, Types of Machine Learning, Supervised Learning Versus Unsupervised Learning Versus Reinforcement Learning, Discriminative Algorithms.

Module II: Feature Engineering: [4L]

Introduction to Data Processing, ETL, Measurement of Purity, Entropy and Gini Index, Normalization and Standardization, Dimension Reduction, ICA (Independent Components Analysis), EM. Mixture of Gaussians, Factor Analysis, Normal Distribution and Gaussian Distribution.

Module III: Classification: [10L]

Introduction to Supervised Learning, Concepts of Linear Algebra, Linear Regression and Logistic Regression, Concepts Bias/ Variance Trade off, Prediction Versus Classification Problem, Naive Bayes, Maximum Entropy, Perceptron, Basic Concept of Neural Network, Generative Learning Algorithms, Gradient Descent, Regularization, Feed Forward Neural Network, Back Propagation Neural Network, Gaussian Discriminant Analysis, Concepts of vectorization , Support Vector Machines, Introduction of Deep Learning, Hidden Markov Model, Genetic Algorithms,

Module IV: Clustering: [12L]

Introduction to Unsupervised learning: Introduction to Clustering, K-means and Hierarchical Clustering, Comparison among classification and clustering, Dimension reduction: PCA (Principal Components Analysis), Factor analysis.

Module V: Machine Learning System Design: [6L]

Underfitting and Overfitting Problem, Bias-Variance as Function of Lambda, Cross Validation, Learning Curves, Error Analysis, Confusion Matrix, Trading off Precision and Recall, ROC Curve, F1-Score and Accuracy Analysis

Module VI: Case studies: [2L]

Applications of ML in Case Studies.

Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books:

Text Books:

1. Bishop, Christopher M. Pattern Recognition and Machine Learning. Springer, 2006.
2. Alpaydin, Ethem. Introduction to Machine Learning. 3rd ed., The MIT Press, 2014.
3. Murphy, Kevin P. Machine Learning: A Probabilistic Perspective. The MIT Press, 2012.
4. Marsland, Stephen. Machine Learning: An Algorithmic Perspective. 2nd ed., CRC Press, 2014.

Reference Books:

1. Shalev-Shwartz, Shai, and Shai Ben-David. Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press, 2014.
2. Géron, Aurélien. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. 2nd ed., O'Reilly Media, 2019.
3. Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed., Springer, 2009.

➠ Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Explain* fundamental machine learning concepts, learning paradigms, and essential principles of supervised, unsupervised, and reinforcement learning.

CO2: *Apply* appropriate feature engineering, preprocessing, and dimensionality reduction techniques, and implement ML algorithms for classification and clustering tasks.

CO3: *Analyze* ML models using bias–variance trade-offs, error analysis, and performance metrics to assess model behaviour and generalization.

CO4: *Evaluate* different ML algorithms, compare model performance, and justify the selection of suitable approaches for specific use cases.

CO5: *Design* and *implement* end-to-end ML systems for real-world problems using systematic workflows, appropriate algorithms, and optimization techniques.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	1	1	1	1	1	-	1	-	-	1	2	1	-
CO2	3	3	2	2	3	-	1	-	1	-	-	1	3	3	1
CO3	2	3	2	3	2	-	1	-	1	-	-	1	3	2	1
CO4	2	3	3	3	3	-	1	-	2	-	-	2	3	3	2
CO5	2	3	3	3	3	1	2	1	2	1	1	3	3	3	3

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

ELECTIVE (G): INTERNET OF THINGS

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Internet of Things	**Course Credit:** 03[3-0-0]
Department: Computer Science Engineering	**Category:** DSE/PEC
Code: XXXXXX	**Semester:** 3^{rd}

➠ Learning Objectives: After completing this course, students will be able to:

1. Understand the fundamental concepts, architecture, and enabling technologies of the IoT.
2. Identify and explain IoT devices, sensors, actuators, and embedded platforms used in real-world applications.
3. Analyze various IoT communication technologies and protocols for device-to-device and device-to-cloud communication.
4. Apply cloud platforms, data management techniques, and analytics for processing and visualizing IoT data.
5. Understand security, privacy, and management challenges in IoT systems and explore emerging trends and applications.

➠ Prerequisite: Basic knowledge of Microprocessor and Micro controller.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Introduction to Internet of Things	4	11%
Module-II: IoT Devices and Embedded Systems	6	17%
Module-III: IoT Communication Technologies and Protocols	6	17%
Module-IV: IoT Data Management and Cloud Integration	8	22%
Module-V: IoT Security, Privacy, and Management	4	11%
Module-VI: IoT Applications, Case Studies, and Emerging Trends	8	22%

➠ Syllabus Outline: .....

Module I: Introduction to Internet of Things: [4L]

Definition and characteristics of IoT, Evolution of IoT and enabling technologies, IoT architecture: layered view (device, gateway, cloud, application), IoT ecosystem and applications: smart homes, smart cities, healthcare, agriculture, industrial IoT; Challenges and limitations of IoT

Module II: IoT Devices and Embedded Systems: [6L]

IoT hardware components: sensors, actuators, microcontrollers; Embedded systems for IoT, Overview of Arduino, Raspberry Pi, ESP8266/ESP32; Sensor data acquisition and actuator control, Power management and energy-efficient devices

Module III: IoT Communication Technologies and Protocols: [6L]

IoT communication models, Short-range communication: RFID, NFC, Bluetooth, ZigBee; Long-range communication: LoRaWAN, NB-IoT, LTE-M; IoT networking protocols: MQTT, CoAP, AMQP, HTTP/REST; Addressing and routing for IoT (IPv6, 6LoWPAN)

Module IV: IoT Data Management and Cloud Integration: [8L]

IoT data lifecycle and data flow, Cloud platforms for IoT: AWS IoT, Azure IoT Hub, Google Cloud IoT; Data storage, processing, and analytics, Edge and fog computing concepts, IoT dashboards and visualization

Module V: IoT Security, Privacy, and Management: [4L]

Security challenges in IoT, Authentication, authorization, and access control; Secure communication and encryption, Privacy issues and data protection, IoT device management and firmware updates

Module VI: IoT Applications, Case Studies, and Emerging Trends: [8L]

Industrial IoT (IIoT), Smart healthcare and wearable IoT, Smart transportation and logistics, Case studies of real-world IoT deployments, Emerging trends: AIoT, Digital Twins, Blockchain for IoT, 5G-enabled IoT

➠ Course Outcome (CO): .....

Upon successful completion of this course, students will be able to:

CO1: *Explain* the concepts, architecture, and components of IoT systems and identify suitable application domains.

CO2: *Analyze* IoT hardware platforms, sensors, and embedded systems used for data acquisition and

control.

CO3: *Apply* appropriate communication technologies and protocols to design IoT-based solutions.

CO4: *Develop* and *integrate* IoT applications with cloud platforms for data storage, processing, and visualization.

CO5: *Assess* security, privacy, and management issues in IoT deployments and evaluate emerging IoT trends and technologies.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	1	-	1	1	-	-	1	1	-	-	1	-	-
CO2	3	2	2	-	2	-	-	-	-	-	1	-	2	-	-
CO3	2	3	3	-	3	-	-	-	-	-	1	-	2	2	-
CO4	2	3	3	2	3	-	-	-	1	-	2	1	2	3	1
CO5	2	3	2	2	2	2	1	-	1	1	2	2	2	2	2

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

ELECTIVE (H): CLOUD COMPUTING

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Cloud Computing	**Course Credit:** 03[3-0-0]
Department: Computer Science Engineering	**Category:** DSE/PEC
Code: XXXXXX	**Semester:** 3^{rd}

➠ Learning Objectives: After completing this course, students will be able to:

1. To Understand fundamentals of cloud computing
2. To acquire good working knowledge of the essentials of Cloud Micro Services

➠ Prerequisite: Knowledge of Python.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Cloud Fundamentals	4	11%
Module-II: Application Architectures	8	22%
Module-III: Cloud Services	6	16%
Module-IV: Cloud Application Development	6	17%
Module-V: Cloud Security	6	17%
Module-VI: Cloud Service Monitoring and Management	6	17%

➠ Syllabus Outline: .....

Module I: Cloud Fundamentals: [4L]

Definition, characteristics, and benefits of Cloud Computing; Cloud service models: IaaS, PaaS, SaaS; Cloud deployment models: Public, Private, Hybrid, Community; Virtualization concepts: Types of hypervisors, containers, VM lifecycle; Cloud economics, scalability, elasticity, multitenancy; Overview of leading cloud platforms (AWS, Azure, GCP)

Module II: Application Architectures: [8L]

Cloud application requirements and design considerations, Service-Oriented Architecture (SOA) and Microservices, Event-driven and serverless architectures (FaaS), Multi-tier cloud application architecture, Scalability patterns: Load balancing, sharding, autoscaling, Design for failure, resilience, fault tolerance in cloud apps

Module III: Cloud Services: [6L]

Compute Services: EC2, Azure VMs, GCP Compute Engine; Storage Services: Object storage, block storage, file storage; Database Services: Relational (RDS), NoSQL (DynamoDB, Cosmos DB); Networking Services: VPC, subnets, gateways, DNS, CDN; Identity & Access Management (IAM) services; Messaging & Queueing: SNS, SQS, Pub/Sub

Module IV: Cloud Application Development: [6L]

Cloud SDKs, APIs, and CLI tools; Deploying applications using PaaS (AWS Elastic Beanstalk, Azure App Service); Docker container development and cloud deployment; Kubernetes fundamentals: Pods, services, deployments; CI/CD pipelines for cloud (GitHub Actions, Azure DevOps, AWS CodePipeline); Serverless development using AWS Lambda, Azure Functions

Module V: Cloud Security: [6L]

Cloud security challenges: Data, network, application, and VM security; Shared Responsibility Model; Authentication, authorization, IAM policies; Encryption at rest and in transit; Secure cloud storage and key management (KMS); Security monitoring, auditing, and compliance (ISO, GDPR, HIPAA basics)

Module VI: Cloud Service Monitoring and Management: [6L]

Cloud monitoring concepts: SLAs, SLOs, SLIs; Performance monitoring tools: CloudWatch, Azure Monitor, Stackdriver; Resource management, autoscaling, and cost optimization; Logging, tracing, incident management; Cloud governance and policy frameworks; Cloud automation with scripts and Infrastructure-as-Code (IaC): Terraform, CloudFormation

Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

Text & Reference Books:

Text Books:

1. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Cloud Computing Principles and Paradigms, 1st Edition, Wiley, 2013.
2. Ronald Krutz and Russell Dean Vines, Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Wiley, 2010.

Reference Books:

1. Toby Velte, Anthony Velte, Robert Elsenpeter, Cloud Computing, A Practical Approach, McGraw Hill, 2010.

➠ Course Outcome (CO): .....

Upon successful completion of this course, students will be able to:

CO1: *Understand* the fundamental concepts, characteristics, architecture, and enabling technologies of cloud computing, including the roles of networks, protocols, and web services.

CO2: *Apply* knowledge of cloud service models (IaaS, PaaS, SaaS, NaaS) and deployment models (Public, Private, Hybrid, Community) to identify suitable solutions for real-world applications.

CO3: *Analyze* the principles of virtualization, resource management, and storage in Infrastructure as a Service (IaaS), including hypervisors, virtual machines, and examples like Amazon EC2.

CO4: *Evaluate* the benefits, challenges, and risks associated with Platform as a Service (PaaS) and Software as a Service (SaaS), incorporating Service-Oriented Architecture (SOA), Web 2.0, and management practices.

CO5: *Design* strategies to address security issues in cloud computing, including network, host, application, and data security, while considering overall challenges and risks in cloud platforms and services.

➠ CO-PO-PSO Mapping: .....

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	-	1	-	2	-	1	-	-	-	2	1	-
CO2	3	3	3	2	2	1	2	1	2	1	1	1	3	3	1
CO3	3	3	2	2	3	1	1	-	1	1	-	1	3	3	1
CO4	2	2	3	3	2	2	2	2	2	2	1	2	2	3	2
CO5	2	3	3	3	2	3	2	1	2	3	1	1	3	2	2

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

OPERATION RESEARCH

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Operation Research	**Course Credit:** 02[2-0-0]
Department: Computer Science Engineering	**Category:** BS
Code: XXXXXX	**Semester:** 3^{rd}

➠ Learning Objectives: After completing this course, students will be able to:

1. Emphasize the application of Operations Research for solving Engineering problems.
2. Understand the meaning, purpose, and tools of Operations Research.
3. Critically analyze a problem, identify, formulate, and solve problems in any engineering field using operations research principles, considering current and future trends.
4. Know and understand common and important engineering problems.
5. Use optimization techniques to enhance systems and to manage enterprise resources using current tools, frameworks, and reusable resources.

➠ Prerequisite: Basic knowledge of mathematics.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: Linear Programming Problems	6	25%
Module-II: Special Types of Linear Programming Problems	2	8%
Module-III: Integer Programming Problems	4	17%
Module-IV: Goal Programming Problems	4	17%
Module-V: Markov Chains	2	8%
Module-VI: Game Theory	6	25%

➠ Syllabus Outline: .....

Module I: Linear Programming Problems: [6L]

An overview and scope of Operations Research and Introduction to Linear Programming (LP) - Illustration of LP Problems - Formulation exercises on LP Problems - Graphical Method of solving LPP - Simplex Method – Unboundedness - Multiple Optimum Solutions - Degeneracy and Cycling Problems - Artificial Variables: Big-M Method - Sensitivity Analysis

Module II: Special Types of Linear Programming Problems: [2L]

Formulation of Transportation Problems - Sensitivity Analysis in Transportation Problems - Assignment Problems

Module III: Integer Programming Problems: [4L]

Formulation, Cutting Plane Method - Branch and Bound Method – Applications

Module IV: Goal Programming Problems: [4L]

Single and Multiple Goal Programming Problems.

Module V: Markov Chains: [2L]

Concepts, Transition Probabilities - Steady-State Probabilities – Applications

Module VI: Game Theory: [6L]

Introduction - Characteristics of Game Theory - Two Person, Zero sum games - Pure strategy - Dominance theory - Mixed strategies - Algebraic and graphical methods.

➠ Teaching–Learning Methodology: .....

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

➠ Text & Reference Books: .....

Text Books:

1. Kanti Swarup, Gupta P.K., and Manmohan, (2008), Operations Research, S. Chand & sons

Reference Books:

1. Hamdy Taha, (1999), Operations Research, PHI.
2. S.D.Sharma, (2006), Operations Research , Kedamanth Ramnath & Co.

3. Hira and Gupta, (2001), Operations Research, S.Chand & Sons.
4. Panneerselvan. R. (2006), Operation Research, Prentice Hall of India Pvt Ltd.

Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Recall* the fundamental concepts of Operations Research and the principles of Linear Programming.

CO2: *Explain* special types of Linear Programming Problems along with their sensitivity analysis.

CO3: *Apply* techniques to appropriate decision-making problems.

CO4: *Analyze* models to handle conflicting objectives in decision environments.

CO5: *Evaluate* models and strategies to determine optimal policies and optimal play in competitive situations.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	1	2	-	-	-	-	-	2	1	-
CO2	3	3	2	2	2	-	2	-	-	-	-	1	3	2	-
CO3	3	3	2	2	2	-	2	1	1	-	1	1	3	3	1
CO4	3	3	3	3	2	-	2	-	1	1	1	2	3	2	1
CO5	3	3	3	3	2	-	2	1	1	1	1	2	3	3	1

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

ACCOUNTING AND MANAGEMENT CONTROL

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Accounting and Management Control	**Course Credit:** 03[2-1-0]
Department: Computer Science Engineering	**Category:** AECC/HSM
Code: XXXXXX	**Semester:** 3^{rd}

➠ Learning Objectives:

1. To enhance the abilities of learners to develop the concept of management accounting and its significance in the business.
2. To enhance the abilities of learners to analyze financial statements.
3. To enable the learners to understand, develop and apply the techniques of management accounting in the financial decision-making in the business corporates.
4. To make the students develop competence with their usage in managerial decision-making and control.

➠ Prerequisite: The primary prerequisite for this course is basic concept of mathematics.

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: The Conceptual Foundations of Control Systems	8	22%
Module-II: The Traditional Instruments of Control in Organizations	8	22%
Module-III: Accountability in Organizations	6	17%
Module-IV: The New Dimensions of Control with Strategies	6	17%
Module-V: Management Control in Specialized Organizations	4	11%
Module-VI: Non-profit organizations	4	11%

Syllabus Outline:

Module I: The Conceptual Foundations of Control Systems: [8L]

Meaning, Nature, and purpose of control systems The new paradigms of Management Control Systems, four elements of control, organizational structure, organizational goals, organizational climate, strategic planning Balancing the four levers of control, balancing the tensions in control systems, six sources of tensions in control systems, opportunities and limitations of the span of control, key control variables, delegation, and decentralization, mutual supportive management systems.

Module II: The Traditional Instruments of Control in Organizations: [8L]

External audit, internal controls, internal audit, role of financial controllers, multiple roles of an auditor, management control process, budgetary control, flexible budget, zero base budget, performance budgeting, master budget, analysis of variance, accounting aspect of control, management audit, marketing and distribution control, different types of audit.

Module III: Accountability in Organizations: [6L]

Dual focus and accountability, differentiate between product costing and accountability, the concept of responsibility centre, management control structure, responsibility accounting, cost centre, profit centre, investment centre, ABC costing, transfer prices, CVP analysis, process control.

Module IV: The New Dimensions of Control with Strategies: [6L]

Behavioral aspect of management control, motivations, morale, participative management, learning curves, HR accounting, knowledge management control, management control with reference to risk management, differentiated controls for different situations, measuring performance to match strategy, balanced score cards.

Module V: Management Control in Specialized Organizations: [4L]

Sectoral applications, controlling the financial sector, the banking sector, the balance sheet concept, the concept of schedule of advances, the use of ABC costing standard, insurance, system of insurance accounts

Module VI: Non-profit organizations: [4L]

Legal environment of non-profit organization, public service organizations, public utility accounts, holding company accounts, government and co-operative business, control in projects, the twelve step process of designing controlling system.

Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

Text & Reference Books:

Text Books:

1. Management Control Systems – Robert N. Anthony & Vijay Govindarajan (McGraw-Hill)
2. Accounting for Management – S. N. Maheshwari & S. K. Maheshwari (Vikas Publishing)
3. Management Accounting – Horngren, Sundem, Stratton, Burgstahler & Schatzberg (Pearson)
4. Cost Accounting: A Managerial Emphasis – Charles T. Horngren (Pearson)
5. Management Control Systems – Kenneth A. Merchant & Wim Van der Stede (Pearson)

Reference Books:

1. Financial Accounting – Jerry J. Weygandt, Paul D. Kimmel, Donald E. Kieso (Wiley)
2. Management Accounting: Principles & Practice – M. N. Arora (Himalaya Publishing House)
3. Strategic Management Accounting – Colin Drury (Cengage)
4. Essentials of Management Control Systems – Leslie G. Eldenburg & Susan K. Wolcott
5. Cost and Management Accounting – T. S. Reddy & Y. Hari Prasad Reddy (Margham Publications)

Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Understand* accounting principles, financial statements, and management control systems

CO2: *Analyze* financial performance using ratio analysis, budgeting, and variance analysis

CO3: *Apply* accounting and control techniques to solve managerial decision-making problems

CO4: *Evaluate* internal control systems and recommend improvements for organizational efficiency

CO5: *Demonstrate* preparation of accounting reports and management control documentation

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	-	-	-	-	-	-	-	-	3	2	1
CO2	2	3	2	2	1	1	-	1	1	-	-	-	3	2	2
CO3	3	2	3	2	2	2	1	2	1	-	-	-	3	3	2
CO4	2	3	2	2	3	2	-	1	2	-	-	-	2	3	2
CO5	3	2	3	2	2	1	1	2	1	-	-	-	3	2	3

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

CUSTOMER RELATIONSHIP MANAGEMENT USING SALESFORCE

➠ Course Information:

School: School of Science & Technology	**Course Type:** L-T-P-S
Name: Customer Relationship Management using Salesforce	**Course Credit:** 02[1-0-2-0]
Department: Computer Science Engineering	**Category:** AECC/HSM
Code: XXXXXX	**Semester:** 3^{rd}

➠ Learning Objectives: After completing this course, students will be able to:

1. Understand the concepts and importance of Customer Relationship Management (CRM).
2. Familiarize students with Salesforce as a leading cloud-based CRM solution.
3. Provide practical experience in customizing and extending CRM functionality.
4. Enable the development and integration of cloud-based business applications.
5. Introduce cloud architecture and the Salesforce platform for scalable enterprise solutions.

➠ Prerequisite: Students should have basic knowledge of Web Technologies, Databases, and Software Engineering

➠ Course Content/ Syllabus Table:

Module No.	No. of Lecture / Contact hour	Weightage (%)
Module-I: CRM and Cloud Computing Basics	4	11%
Module-II: Salesforce Platform Overview	4	11%
Module-III: Declarative Development and Automation	6	17%
Module-IV: Programmatic Development with Apex	8	22%
Module-V: Lightning Web Components and App Development	6	17%
Module-VI: Integration and Analytics	8	22%

➠ Syllabus Outline: .....

Module I: CRM and Cloud Computing Basics : [4L]

- Introduction to CRM: Evolution, types, and business relevance
- CRM components: Operational, Analytical, Collaborative
- Introduction to Cloud Computing: IaaS, PaaS, SaaS
- CRM as SaaS: Overview of market leaders

Module II: : Salesforce Platform Overview : [4L]

- Salesforce architecture and editions
- Navigating Salesforce Lightning interface
- Standard objects and their relationships (Account, Contacts, Leads)
- Custom objects and schema builder

Module III: Declarative Development and Automation: [6L]

- Customizing page layouts and record types
- Validation rules, workflow rules, and process builder
- Flows and approval processes
- Salesforce App Builder and Lightning App setup

Module IV: Programmatic Development with Apex: [8L]

- Apex syntax and data types
- SOQL and SOSL queries
- Writing triggers and classes
- Introduction to asynchronous Apex (future, batch, queueable)

Module V: Lightning Web Components and App Development: [6L]

- Component-driven development with LWC
- Events and data binding
- Creating reusable components
- Deployment using change sets and unmanaged packages

Module VI: Integration and Analytics: [8L]

- REST API and external integrations
- AppExchange and prebuilt integrations
- Reports, dashboards, and Einstein Analytics
- Final project: CRM application for a fictional business

Teaching–Learning Methodology:

- ***Pedagogy for Course Delivery:*** Hybrid Mode (Offline Class/ Presentation/ Video/ MOOC)
- ***Continuous Assessment:*** Quiz/ Assessment/ Presentation/ Problem solving etc.

Text & Reference Books:

Text Books:

1. Paul Battisson, Learning Salesforce Development with Apex, Packt Publishing
2. Salesforce Official Documentation: https://developer.salesforce.com/docs

Reference Books:

1. Siddhesh Kabe, Salesforce Essentials for Administrators, Packt Publishing
2. Michael Wicherski, Advanced Apex Programming, Salesforce Press
3. Trailhead (Salesforce's Official Learning Platform): https://trailhead.salesforce.com

Course Outcome (CO):

Upon successful completion of this course, students will be able to:

CO1: *Explain* the fundamentals, architecture, and evolution of Customer Relationship Management (CRM) systems with emphasis on the Salesforce ecosystem.

CO2: *Apply* Salesforce CRM features to manage sales, service, and marketing workflows in real-world business scenarios.

CO3: *Design* and *implement* process automation using validation rules, workflow rules, and Flow Builder to streamline business operations.

CO4: *Analyze* Salesforce reports and dashboards to extract actionable business insights and support decision-making.

CO5: *Able* to develop and *demonstrate* a mini-project simulating a real-world CRM scenario and prepare for Salesforce Administrator (ADM 201) certification.

CO-PO-PSO Mapping:

Course Outcomes	Programme Outcomes												Programme Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	1	1	-	1	-	-	-	1	1	-	-	1	-	-
CO2	2	2	2	-	2	-	-	-	-	-	1	-	2	2	-
CO3	2	2	3	-	3	-	-	-	-	-	1	-	2	3	-
CO4	3	3	3	1	3	-	-	1	1	-	1	1	3	3	1
CO5	3	3	3	2	3	-	-	2	2	1	2	2	3	3	3

1: Slight (Low) *2: Moderate (Medium)* *3: Substantial (High)*

ARTIFICIAL INTELLIGENCE LAB

Course Information:

School: School of Science & Technology	**Course Type:** L-T-P
Name: Artificial Intelligence Lab	**Course Credit:** 01[0-0-2]
Department: Computer Science Engineering	**Category:** CC/PCC
Code: XXXXXX	**Semester:** 3^{rd}

List of Practicals: .....

1. Execute the Basic Operations of SWI Prolog and Python along with the installation process of Python Jupyter Notebook and SWI Prolog
2. Implementation of relational tree structure in SWI Prolog
3. Implementation of Circuit Design Logic Using SWI Prolog
4. Implementation of Predecessors and Successors in SWI Prolog
5. Implementation of Graph Colouring(Vertices, Edges, Regions) in SWI Prolog
6. Implementation of Greedy Algorithm using Python
7. Hill Climbing and A* Algorithm using Python
8. Implement BFS and DFS using Python
9. Implement the Tower of Hanoi using SWI Prolog and Python
10. Implement BFS and DFS using Python
11. 4 Queens Problem using Python
12. Basic implementations and innovative algorithm design using of 2 Fuzzy Sets like Union, Intersection, Negation etc. using Python
13. Case Study with Analysis

SEMESTER-IV

Sl. No.	Paper Name	Code	Category	Credit	Type			
					L	T	P	S
1	Major Project		SEC/PSE	12	0	0	0	24
2	Grand Viva		SEC/PSE	6	0	0	0	0
3	Foreign Language – I (German/ Spanish/ Japanese)		USC/MUS	2	2	0	0	0
Total Credit=20