Sister Nivedita University

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

School of Engineering
Department of Computer Science & Engineering

SYLLABUS
FOR
MASTER OF COMPUTER APPLICATION (MCA)

Regulations (R24) [NEP]

R24–25 Academic Session

Credit Definition

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

Total Credit Distribution

Semester	Credits						Credits/ Semester
	BS	HSM	MUS	PC	PE	PSE	
1	3	3	2	15	0	0	23
2	0	3	2	19	0	0	24
3	2	3	2	8	2	6	23
4	0	0	2	0	0	18	20
Credits/ Course	**5**	**9**	**8**	**42**	**2**	**24**	**90**

Category Definition

Definition of Category/ Type	Code	No	Credit
Basic Science	BS	1	05
Engineering Science	ES	2	00
Professional Core	PC	3	42
Professional Elective (Discipline Specific)	PE	4	02
Humanities & Social Science including Management	HSM	6	09
Project Work/ Seminar/ Internship/ Entrepreneurship	PSE	7	24
Mandatory/ University Specified (Environmental Sc./ Induction Training/ Indian Constitution/ Foreign language)	MUS	8	08
Total			**90**

Program Outcomes (POs):

PO1. Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and engineering specialization to the solution of complex engineering problems.

PO2. Problem analysis: Identify, formulate, research literature, and analyse engineering problems to arrive at substantiated conclusions using first principles of mathematics, natural and engineering sciences.

PO3. Design/Development of solutions: Design solutions for complex engineering problems and design system components, processes to meet the specifications with consideration for the public health and safety and the cultural societal and environmental considerations.

PO4. Conduct investigations of complex problems: Use research-based knowledge including design of experiments, analysis and interpretation of data and synthesis of the information to provide valid conclusions.

PO5.Modern tool usage: Create, select and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

PO6. The engineer and society: Apply reasoning informed by the contextual knowledge to access societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of and need for sustainable development.

PO8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9. Individual and team work: Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

PO10. Communications: Communicate effectively with the engineering community and with the society at large. Be able to comprehend and write effective reports documentation. Make effective presentations and give and receive clear instructions.

PO11. Project management and finance: Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments.

PO12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

FIRST YEAR

SEMESTER-I

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

SEMESTER-II

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

SECOND YEAR

SEMESTER-III

Sl. No.	Paper Name	Category	Credit	Type		
				L	T	P
1	Artificial Intelligence	PC	3	0	0	0
2	Theory of Computation	PC	3	0	0	0
3	Elective	PC	2	2	0	0
4	Operation Research	BS	2	0	0	0
5	Accounting and Management Control	HSM	3	2	1	0
6	Foreign Language – III (German/ Spanish/ Japanese)	MUS	2	2	0	0
7	Minor Project-I	PSE	2	0	0	2
Practical						
8	Artificial Intelligence Lab	PC	2	0	0	2
			Total Credit=23			

Elective (2204123)

A. Data Warehousing and 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	Category	Credit	Type		
				L	T	P
1	Major Project (6 months)	PSE	12	0	0	0
2	Grand Viva	PSE	6	0	0	0
3	Foreign Language – I (German/ Spanish/ Japanese)	MUS	2	2	0	0
			Total Credit=20			

FIRST YEAR

SEMESTER-I

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

COMPUTER ORGANIZATION AND ARCHITECTURE

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Computer Organization and Architecture	**COURSE CREDIT:** 03[2-1-0]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 1^{st}

THEORY

Learning objectives:

1) Illustrate the structure, function, and characteristics of computer systems.
2) Exhibit the design of the various functional units of digital computers.
3) Discuss different types of memories and their properties.
4) Introduce basics of Parallel Computer Architecture.

Prerequisite: Boolean algebra, Machine Instructions, Dataflow.

Course content/Syllabus table:

Module No.	No. of lecture / Contact hour	Weightage (%)
Module-I: Principles of Computer design	6	20%
Module-II: Different types of Instructions	6	13%
Module-III: CPU Designing	8	13%
Module-IV: Memory	8	13%
Module-V: Input-output Organization	4	24%
Module-VI: Type module name	4	17%

SYLLABUS OUTLINE:

Module I: Principles of Computer design: [6L]

Software, hardware interaction layers in computer architecture, Central processing unit.

Module II: Different types of Instructions: [6L]

Machine language instructions, addressing modes, instruction types, Instruction set selection, Instruction cycle and execution cycle.

Module III: CPU Designing: [8L]

Control unit, Data path and control path design, Microprogramming Vs hardwarecontrol, RISC Vs. CISC, Pipelining in CPU design: Superscalar processors.

Module IV: Memory: [8L]

Memory system, Storage technologies, Memory array organization, Memory hierarchy, interleaving, cache and virtual memories and architectural aids to implement these.

Module V: Input-output Organization: [4L]

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

Module VI: Performance evaluation: [4L]

SPEC marks, Transaction Processing benchmarks

- ***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

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2	1	-	-	-	-	-	-	-	1
CO2	-	3	2	2	-	-	-	-	-	-	-	-
CO3	2	3	1	1	-	-	-	-	-	-	-	1
CO4	2	3	-	-	3	-	-	-	-	-	2	-
CO5	1	1	3	3	-	-	-	-	-	-	1	1
CO6	2	1	1	2	-	-	-	-	-	-	-	1
Avg	2	2.16	2	1.8	2	-	-	-	-	-	-	1.2

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: Understand the structure, function and characteristics of computer systems and understand the design of the various functional units and components of computers.

CO2: Design the arithmetic and Logic unit and understand the floating- and fixed-point number representation

CO3: Analyze the performance of ripple carry adder and carry look ahead adder and understand the multiplication and division algorithm

CO4: Identify the elements of control unit and design of control unit

CO5: Explain the function of each element of a memory hierarchy.

CO6: Understand the input output subsystem and analyze the role of interrupts in process state transition.

DATA STRUCTURES

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Data Structures	**COURSE CREDIT:** 03[2-1-0]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 1^{st}

THEORY

Learning objectives:

1) To impart the basic concepts of data structures and algorithms.
2) To understand concepts about searching and sorting techniques.
3) To understand basic concepts about stacks, queues, lists, trees and graphs.
4) To understanding about writing algorithms and step by step approach in
5) solving 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: [L]

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: [L]

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: [L]

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

Module VI: Searching and Hashing: [L]

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

- ***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

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1									
CO2	3	3	3	1								1
CO3	3	3	3	2								1
CO4	3	3	3	1								1
CO5	3	3	3	2								1
CO6	3	3	3	2								1
Avg.	3	2.83	2.66	1.33								0.83

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: To be able to classify linear and non-linear data structure.

CO2: To be able to solve different problems using Arrays.

CO3: To be able to make use of linked list for various operations on polynomials, sparse matrix etc.

CO4: To be able to utilize the knowledge of Stack, Queues in solving real life problem.

CO5: To be able to apply the knowledge of several binary trees in problem solving.

CO6: To be able to identify of the most appropriate searching or sorting algorithm for enhancing the efficiency (i.e. reduce the run-time) or for better memory utilization.

DATABASE MANAGEMENT SYSTEM

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Database Management System	**COURSE CREDIT:** 03[2-1-0]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 1^{st}

THEORY

Learning objectives:

1) To understand the basic concepts and the applications of database systems
2) To be master the basics of SQL and construct queries using SQL
3) To understand the relational database design principles
4) To become familiar with the basic issues of transaction processing and concurrency control
5) To 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	10%
Module-II: Data Models	6	20%
Module-III: Database Design, ER-Diagram and Database Language	10	20%
Module-IV: Relational Algebra and Relational Calculus	10	20%
Module-V: Constraints, Views and SQL	6	10%
Module-VI: Indexing and Transactions	12	20%

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: [6L]

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: [10L]

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: [10L]

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: [12L]

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.

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: To be able to discuss basic concepts, data models, types of users and appreciate the applications of database systems.

CO2: To be able to understand the logical design of the database including E-R models and the concept of generalization, specialization and aggregation.

CO3: To be able to apply with a relational database system and Normalization.

CO4: To be able to explain the basic concepts of relational database design, relational algebra and SQL.

CO5: To be able to analyze relational database and formulate SQLqueries on data.

CO6: To be able to describe transaction processing and concurrency control concepts.

DISCRETE MATHEMATICS

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Discrete Mathematics	**COURSE CREDIT:** 03[2-1-0]
DEPARTMENT: Computer Science	**CATEGORY:** BS
CODE: XXXXXX	**SEMESTER:** 1^{st}

THEORY

Learning objectives: On completion of the course, student will be able to: 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 and 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 deductionsystemand axiomsystem;Soundness and completeness. Distributive and complete lattices.

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: Understand the fundamentals of Propositional Logic

CO2: Identify truth tables and logical operators to analyse problems.

CO3: Understand the fundamental theorems of Group theory.

CO4: Understand the fundamental concepts in graph theory.

CO5: Apply the knowledge of Boolean algebra in switching circuits.

CO6: Use Max-flow Min-cut theorem, Ford and Fulkerson algorithm to design complex engineering problems.

COMMUNICATIVE ENGLISH

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Communicative English	**COURSE CREDIT:** 03[2-1-0]
DEPARTMENT: Computer Science	**CATEGORY:** HSM
CODE: XXXXXX	**SEMESTER:** 1st

THEORY

Learning objectives:

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	%
Module-II: Slide preparation and oral presentation principles	6	%
Module-III: Written presentation of technical material	6	%
Module-IV: Preparation of bibliography, basic of official correspondence	6	%
Module-V: Preparation of bio-data	6	%
Module-VI: Group discussions should also be used and feedback is given to students	6	%

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

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: To sensitize students to the language, forms and types of poetry, fiction, prose, film and drama

CO2: To sensitize students to the nuances of spoken and written forms of English

CO3: To familiarize students with the concepts of copy-editing and impart to them basic copy-editing skills and familiarize them with the diverse concerns addressed by feminism.

CO4: To update and expand basic informatics skills and attitudes relevant to the emerging knowledge society

CO5: To enable them to produce grammatically and idiomatically correct language and help master writing techniques to meet academic and professional needs.

CO6: To provide sufficient practice in Vocabulary, Grammar, Comprehension, and Remedial English from the perspective of career-oriented tests.

COMPUTER ARCHITECTURE AND ORGANIZATION LAB

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Computer Architecture and Organization Lab	**COURSE CREDIT:** 02[0-0-4]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 1^{st}

List of practical:

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 Engineering	**COURSE TYPE:** L-T-P
NAME: Data Structures Lab	**COURSE CREDIT:** 02[0-0-4]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 1^{st}

List of practical:

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 Engineering	**COURSE TYPE:** L-T-P
NAME: Database Management System Lab	**COURSE CREDIT:** 02[0-0-4]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 1^{st}

List of practical:

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	Category	Credit	Type		
				L	T	P
1	Software Engineering using UML	PC	3	3	0	0
2	Computer Networks	PC	3	0	0	0
3	Python Programming	PC	3	3	0	0
4	Operating Systems	PC	4	4	0	0
5	Management Information System	HSM	3	3	0	0
6	Foreign Language – II (German/ Spanish / Japanese)	MUS	2	2	0	0
Practical						
7	Software Engineering using UML Lab	PC	2	0	0	2
8	Python Programming Lab	PC	2	0	0	2
9	Operating Systems Lab	PC	2	0	0	2
Total Credit=24						

SOFTWARE ENGINEERING USING UML

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Software Engineering using UML	**COURSE CREDIT:** 03[2-1-0]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 2^{nd}

THEORY

Learning objectives:

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	8%
Module-II: Requirements Analysis Using Case Modeling	8	21%
Module-III: Transfer from Analysis to Design in the Characterization Stage, Finding objects from Flow of Events	8	25%
Module-IV: The Logical View Design Stage: The Static Structure Diagrams	8	21%
Module-V: Package Diagram Model and Dynamic Model: State Diagram / Activity Diagram	4	13%
Module-VI: Component Diagram Model and Deployment Model	4	13%

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.

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: Identify the purpose and methods of use of common object-oriented design patterns.

CO2: Select and apply these patterns in their own designs for simple programs.

CO3: Represent the data dependencies of a simple program using UML.

CO4: Represent user and programmatic interactions using UML.

CO5: Create design documentation outlining the testable and complete design of a simple program.

CO6: Produce and present documents for the purpose of capturing software requirements and specification.

COMPUTER NETWORKS

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Computer Networks	**COURSE CREDIT:** 03[2-1-0]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 2^{nd}

THEORY

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	8%
Module-II: Physical layer and Media	10	21%
Module-III: Data Link Layer and Medium Access Sub Layer	12	25%
Module-IV: Network Layer	10	21%
Module-V: Transport Layer	6	13%
Module-VI: Application Layer	6	13%

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: [10L]

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: [12L]

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/C HDLC, Ethernet, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN.

Module IV: Network Layer: [10L]

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: [6L]

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

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: To be able to understand data communication components, representation of data, physical topologies, and protocols.

CO2: To be able to understand Analog and Digital transmission, multiplexing and working of transmission media.

CO3: To be able to solve problems related to error correction/detection and protocols of media access control layer.

CO4: To be able to solve IP subnetting problems and routing problems.

CO5: To analyze basic operations of transport layer and congestion control mechanisms.

CO6: To be able to understand about various application layer functionalities.

PYTHON PROGRAMMING

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Python Programming	**COURSE CREDIT:** 03[2-1-0]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 2^{nd}

THEORY

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	10	21%
Module-II: Control Structures	10	21%
Module-III: Collections	10	21%
Module-IV: Strings and Regular Expressions	10	21%
Module-V: Functions	4	8%
Module-VI: Handling Exceptions and File Handling	4	8%

SYLLABUS OUTLINE:

Module I: Introduction to Python Programming: [10L]

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: [10L]

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: [10L]

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: [10L]

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

Module V: Functions: [4L]

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: [4L]

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

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: Interpret the basic representation of the data structures and sequential programming

CO2: Knowledge of, and ability to use control framework terminologies.

CO3: Ability to work out using the core data structures as lists, dictionaries, tuples, and sets.

CO4: Choose appropriate programming paradigms, interrupt and handle data using files to propose solution through reusable modules.

CO5: Propose possible error-handling constructs for unanticipated states/inputs

CO6: Implement exemplary applications on the real-world problems.

OPERATING SYSTEMS

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Operating Systems	**COURSE CREDIT:** 04[3-1-0]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 2^{nd}

THEORY

Learning objectives: This course OPERATING SYSTEMS is an essential part of any Computer-Science education. The purpose of this course is 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: Good knowledge of C, Computer Organization and Architecture, x86 Assembly level programming.

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.

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: To be able to understand the design of an operating system and its types. I/O structures and storage structures.

CO2: To be able to apply process scheduling algorithm in various batch process scheduling scenarios.

CO3: To be able to solve process synchronization, and deadlock avoidance problems.

CO4: To be able to compare different memory and I/O management approaches and use system calls for managing processes, memory and the file system.

CO5: To be able to understand the structure and organization of the file system.

CO6: To be able to compare and use different Disk scheduling techniques.

MANAGEMENT INFORMATION SYSTEMS

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Management Information Systems	**COURSE CREDIT:** 03[0-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** HSM
CODE: XXXXXX	**SEMESTER:** 2^{nd}

THEORY

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.

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: Relate the basic concepts and technologies used in the field of management information systems.

CO2: Compare the processes of developing and implementing information systems.

CO3: Outline the role of the ethical, social, and security issues of information systems.

CO4: Translate the role of information systems in organizations, the strategicmanagement processes, with the implications for the management.

CO5: Apply the understanding of how various information systems like DBMS worktogether to accomplish the information objectives of an organization.

CO6: Application of the theoretical models used in database management systems toanswer business questions.

SOFTWARE ENGINEERING USING UML LAB

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Software Engineering using UML Lab	**COURSE CREDIT:** 02[0-0-4]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 2^{nd}

List of practical:

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 needsto 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 Engineering	**COURSE TYPE:** L-T-P
NAME: Python Programming Lab	**COURSE CREDIT:** 02[0-0-4]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 2^{nd}

List of practical:

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 Engineering	**COURSE TYPE:** L-T-P
NAME: Operating Systems Lab	**COURSE CREDIT:** 02[0-0-4]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 2^{nd}

List of practical:

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	Category	Credit	Type		
				L	T	P
1	Artificial Intelligence	PC	3	0	0	0
2	Theory of Computation	PC	3	0	0	0
3	Elective	PC	2	2	0	0
4	Operation Research	BS	2	0	0	0
5	Accounting and Management Control	HSM	3	2	1	0
6	Foreign Language – III (German/ Spanish/ Japanese)	MUS	2	2	0	0
7	Minor Project-I	PSE	2	0	0	2
Practical						
8	Artificial Intelligence Lab	PC	2	0	0	2
			Total Credit=23			

Elective (2204123)

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 Engineering	**COURSE TYPE:** L-T-P
NAME: Artificial Intelligence	**COURSE CREDIT:** 03[3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 3^{rd}

THEORY

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.

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

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

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: To be able to understand the informed & uninformed problems and apply search strategies to solve.

CO2: To be able to apply difficult real-life problems in a state space representation so as to solve them using AI techniques like searching and game playing.

CO3: To be able to design and evaluate intelligent expert models for perception and prediction from intelligent environment.

CO4: To be able to Identify valid solutions for problems involving uncertain inputs or outcomes by using decision making techniques

CO5: To be able to demonstrate and enrich knowledge to select and apply AI tools to synthesize information and develop models within constraints of application area.

CO6: To be able to analyse the issues involved in knowledge bases, reasoning systems and planning

THEORY OF COMPUTATION

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Theory of Computation	**COURSE CREDIT:** 03[3-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 3^{rd}

THEORY

Learning objectives: This course focuses on the basic theory of Computer Science and formal methods of computation like automata theory, formal languages, grammars and Turing Machines. The objective of this course is to explore the theoretical foundations of computer science from the perspective of formal languages and classify machines by their power to recognize languages.

Prerequisite: The primary prerequisite for this course is reasonable "mathematical sophistication." The basic mathematical notations are required to know. The logical functional principles of machine are also need to know. Sets & Types, Sequences, Tuples, Propositional and Predicate Logic, Mathematical Induction, Recursive Definitions, Big-O Notation, Relations and Functions

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, tran-

sition 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.

- ***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.
3) Papadimitrou, C. and Lewis, C.L., "Elements of the Theory of Computation", 2nd edition, Pearson/Prentice Hall India, 2009.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3		2	2								1
CO2	3			1								1
CO3	3	2	1			1						
CO4		2	2	1								1
CO5	2			1	2	1						1
CO6	2	2	2			1						
Avg.	2.16	1	1.16	0.83	0.33	0.5						0.66

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: To be able to Understand the fundamental concepts of Finite State Machines and Models

CO2: To be able to Understand the fundamental concepts of Formal Languages and Automata.

CO3: To be able to apply the pumping lemma, closure properties to problems.

CO4: To be able to Understand the fundamental concepts of Context free grammars.

CO5: To be able to Understand the fundamental concepts of Pushdown Automata.

CO6: To be able to Understand the fundamentals of Turing machine and Linear Bounded Automata.

ELECTIVE (A): DATA WAREHOUSING AND DATA MINING

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Data Warehousing and Data Mining	**COURSE CREDIT:** 02[2-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** PE
CODE: XXXXXX	**SEMESTER:** 3^{rd}

THEORY

Learning objectives:

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 like social, scientific, and environmental contexts.
4) Develop skills in selecting the appropriate data mining algorithm for solving practical problems.

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: Data warehousing, business analysis and on-line analytical processing (OLAP)	6	%
Module-II: Data mining introduction	6	%
Module-III: Data mining frequent pattern analysis	6	%
Module-IV: Classification	6	%
Module-V: Clustering	6	%
Module-VI: WEKA tool	6	%

SYLLABUS OUTLINE:

Module I: Data warehousing, business analysis and on-line analytical processing (OLAP): [6L]
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: Module Name: [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: [6L]

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: [6L]

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: [6L]

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.

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: Design Data Warehouses to solve real-world problems.

CO2: Assess the raw input data, and process it to provide suitable input for a range of data mining algorithms.

CO3: Discover and measure interesting patterns from different kinds of databases.

CO4: Evaluate and select appropriate data mining algorithms and apply, and interpret and report the output appropriately.

CO5: Understand and deploy appropriate classification and clustering techniques.

CO6: Implement the Data Mining techniques to conceptualize a Data Mining solution to a practical problem.

ELECTIVE (B): COMPILER DESIGN

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Compiler Design	**COURSE CREDIT:** 02[2-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** PE
CODE: XXXXXX	**SEMESTER:** 3^{rd}

THEORY

Learning objectives: On completion of the course, student will be able to: understand the structure of a compiler, and how the source and target languages influence various choices in its design, 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. Students will also 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		%
Module-II: Lexical Analysis		%
Module-III: Syntax Analysis		%
Module-IV: Syntax directed translation and Type		%
Module-V: Run time environments and Intermediate Code Generation		%
Module-VI: Code optimization and Code generations		%

SYLLABUS OUTLINE:

Module I: Introduction to Compiling: [L]

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

Module II: Lexical Analysis: [L]

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: [L]

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: [L]

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: [L]

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: [L]

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.

- ***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

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: To identify different phases and passes of the compiler and also to use the compiler tools

CO2: To able to analyze and compare different types of compiler tools to meet the requirements of the realistic constraints of compilers.

CO3: To understand the parser and its types i.e. Top-Down and Bottom-up parsers and construction of LL, SLR, CLR, and LALR parsing table and evaluate the issues

CO4: To Construct the compiler using the syntax-directed translation method and get knowledge about the synthesized and inherited attributes

CO5: To collect knowledge about run-time data structure like symbol table organization and different techniques used in that

CO6: To understand the target machine's run time environment, its instruction set for code generation, and techniques used for code optimization

ELECTIVE (C): DISTRIBUTED DATABASE SYSTEM

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Distributed Database System	**COURSE CREDIT:** 02[2-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** PE
CODE: XXXXXX	**SEMESTER:** 3^{rd}

THEORY

Learning objectives:

1) The purpose of the course is to 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) Topics include distributed DBMS architecture and design; 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		%
Module-II: Logical Time		%
Module-III: Message Ordering & Snapshots		%
Module-IV: Distributed Mutex & Deadlock		%
Module-V: Recovery & Consensus		%
Module-VI: P2P & Distributed Shared Memory		%

SYLLABUS OUTLINE:

Module I: Introduction: [L] 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: [L] A framework for a system of logical clocks Scalar time Vectortime Physical clock synchronization: NTP.

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

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: Fragment a database both horizontally and vertically for optimal performance.

CO2: Allocate replicas of fragments for best performance.

CO3: Optimize queries for optimal performance across a distributed database.

CO4: Add distributed transaction management control including concurrency control and replica control to a distributed database.

CO5: Demonstrate expertise in reading peer-reviewed papers in distributed databases and explaining them in writing.

CO6: Discuss how current database products implement database distribution including query optimization.

ELECTIVE (D): AI AND NEURAL NETWORK

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: AI and Neural Network	**COURSE CREDIT:** 02[2-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** PE
CODE: XXXXXX	**SEMESTER:** 3^{rd}

THEORY

Learning objectives:

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: Basic computer knowledge and Data Structure and Algorithm.

Course content/Syllabus table:

Module No.	No. of lecture / Contact hour	Weightage (%)
Module-I: Introduction		%
Module-II: Searching		%
Module-III: Knowledge Representation		%
Module-IV: Characteristics of Neural Networks, Historical Development of Neural Networks Principles, Artificial Neural Networks		%
Module-V: Feedforward Neural Networks		%
Module-VI: Competitive Learning Neural Networks & Complex pattern Recognition		%

SYLLABUS OUTLINE:

Module I: Introduction: [L]

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: [L]

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: [L]

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: [L]

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: [L]

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: [L]

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

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: Understand the difference between a biological neuron and an artificial neuron. **CO2:** Understand the application areas of neural networks.

CO3: Understand the building blocks of Neural Networks.

CO4: Develop neural network models.

CO5: Design and develop applications using neural networks.

CO6: Learn to design and build neural network models and develop learning algorithms for machine learning.

ELECTIVE (E): CRYPTOGRAPHY AND NETWORK SECURITY

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Cryptography and Network Security	**COURSE CREDIT:** 02[2-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** PE
CODE: XXXXXX	**SEMESTER:** 3^{rd}

THEORY

Learning objectives: On completion of the course, student 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		%
Module-II: Cryptography: Concepts & Techniques		%
Module-III: Symmetric Key Algorithm		%
Module-IV: Asymmetric Key Algorithm, Digital Signature and RSA		%
Module-V: Internet Security Protocols, User Authentication		%
Module-VI: Electronic Mail Security and Firewall		%

SYLLABUS OUTLINE:

Module I: Attacks on Computers & Computer Security: [L]
Introduction, Need for Security, Security approaches, Principles of Security, Types of attack

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

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

Module III: Symmetric Key Algorithm: [L]
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: [L]
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: [L]
Basic Concepts, SSL protocol, Authentication Basics, Password, Authentication Token, Certificate based Authentication, Biometric Authentication.

Module VI: Electronic Mail Security and Firewall: [L]
Basics of mail security, Pretty Good Privacy, S/MIME, Introduction to Firewall, Types of firewall, Firewall Configurations, DMZ Network

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: To understand the fundamental of attacks and the need of security.

CO2: To be able to secure a message over insecure channel by various means.

CO3: Have a strong understanding of different cryptographic algorithms and techniques and be able to use them.

CO4: To learn about how to maintain the Confidentiality, Integrity and Availability of a data.

CO5: To understand various protocols for network security to protect against the threats in the networks.

CO6: To apply methods for authentication, access control, intrusion detection and prevention. Identify and mitigate software security vulnerabilities in existing systems.

ELECTIVE (F): MACHINE LEARNING

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Machine Learning	**COURSE CREDIT:** 02[2-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** PE
CODE: XXXXXX	**SEMESTER:** 3^{rd}

Module No.	No. of lecture / Contact hour	Weightage (%)
Module-I: Introduction to Machine Learning	4	10%
Module-II: Feature Engineering	6	20%
Module-III: Classification	8	20%
Module-IV: Clustering	8	20%
Module-V: Machine Learning System Design	6	10%
Module-VI: Case studies	4	20%

SYLLABUS OUTLINE:

Module I: Introduction to Machine Learning: [4L]

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

Module II: Feature Engineering: [7L]

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: [8L]

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: [7L]

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 : [4L]
Applications of ML in Case Studies.

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	1								1
CO2	3	2		2	1							2
CO3	2	2	3	2	2							2
CO4	1	2	3	3	2							3
CO5	2	2	3	3	3							3
CO6		2	3	3	2							
Avg.	2.20	2.17	2.80	2.33	2.0							2.20

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: To be able to discuss the basics of learning problems with hypothesis

CO2: To be able to understand the features of machine learning to deal with real world problems

CO3: To be able to differentiate the machine learning algorithms as supervised learning and unsupervised learning

CO4: To be able to design and analyze various classification and clustering algorithms

CO5: To be able to develop and tune the machine learning models with datasets

CO6: To be able to evaluate the models for optimization engineering problems

ELECTIVE (G): INTERNET OF THINGS

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Internet of Things	**COURSE CREDIT:** 02[2-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** PE
CODE: XXXXXX	**SEMESTER:** 3^{rd}

THEORY

Learning objectives:

1) To understand basic principles and concepts of Internet-of-Things use cases, applications, architecture and technologies.
2) To get an overview of an end-to-end IoT system encompassing the edge, cloud and application tiers

Prerequisite: Basic knowledge of Microprocessor and Micro controller.

Course content/Syllabus table:

Module No.	No. of lecture / Contact hour	Weightage (%)
Module-I: Introduction to IoT and Use cases		%
Module-II: Architecture		%
Module-III: Sensors		%
Module-IV: Industrial Systems		%
Module-V: Networking and Communication for IoT		%
Module-VI: Network protocols		%

SYLLABUS OUTLINE:

Module I: Introduction to IoT and Use cases: [L]
Understanding basic concepts of IoT, Consumer IoT vs Industrial Internet, Fundamental building blocks, Use Cases of IoT in various industry domains.

Module II: Architecture: [L]
IoT reference architectures, Industrial Internet Reference Architecture, Edge Computing, IoT Gateways, Data Ingestion and Data Processing Pipelines, Data Stream Processing.

Module III: Sensors: [L]

Introduction to sensors and transducers, integrating sensors to sensor processing boards

Module IV: Industrial Systems: [L]

Introduction to industrial data acquisition systems, industrial control systems and their functions

Module V: Networking and Communication for IoT: [L]

Recap of OSI 7 layer architecture and mapping to IoT architecture, Introduction to proximity networking technologies (ZigBee, Bluetooth, Serial Communication)

Module VI: Network protocols: [L]

Industrial network protocols (Modbus, CANbus), Communicating with cloud applications (web services, REST, TCP/IP and UDP/IP sockets, MQTT, WebSockets, protocols. Message encoding (JSON, Protocol Buffers). Time SeriesData and their characteristics, time series databases, basic time series analytics, data summarization and sketching, dealing with noisy and missing data, anomaly and outlier detection.

- ***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) Samuel Greengard, The Internet of Things, MIT Press Essential Knowledge Series, 2015

Reference Books:

1) Ben Fry, Visualizing Data-Exploring and Explaining Data with the Processing Environment, O'Reilly Media, 2008.
2) Andrew K Dennis, Raspberry Pi Computer Architecture Essentials, Packt Publishing, 2016

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: Understand basic principles and concepts of Internet-of-Things use cases, applications.

CO2: Understand basic concepts of Architecture of IoT.

CO3: Describe Sensor and Industrial systems.

CO4: Understand Networking and communication for IoT.

CO5: Comprehend IoT data processing and storage.

CO6: Demonstrate IoT applications in various domains using prototype models

ELECTIVE (H): CLOUD COMPUTING

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Cloud Computing	**COURSE CREDIT:** 02[2-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** PE
CODE: XXXXXX	**SEMESTER:** 3^{rd}

THEORY

Learning objectives:

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

Prerequisite: Have the knowledge of Python.

Course content/Syllabus table:

Module No.	No. of lecture / Contact hour	Weightage (%)
Module-I: Cloud Fundamentals		%
Module-II: Application Architectures		%
Module-III: Cloud Services		%
Module-IV: Cloud Application Development		%
Module-V: Cloud Security		%
Module-VI: Cloud Service Monitoring and Management		%

SYLLABUS OUTLINE:

Module I: Cloud Fundamentals: [L]

Cloud Service Components - Deployment Models – Application of Cloud Computing

Module II: Application Architectures: [L]

Monolithic and Distributed – Micro Service fundamentals – Design Approach – Cloud Native Application – Application Integration Process – API fundamental – API Management

Module III: Cloud Services: [L]

Application Services - Deployment and Management Services - Amazon Web Services - Windows Azure

Module IV: Cloud Application Development: [L]

Python-Refresher, Use cases

Module V: Cloud Security: [L]

Security Basics and Benefits – Challenges

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

Cloud Security Monitoring Tools

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	3	1	-	-	-	-	-	-	-
CO2	3	3	3	2	1	-	-	-	-	-	-	1
CO3	3	3	3	2	1	-	-	-	-	-	-	-
CO4	3	3	3	2	1	-	-	-	-	-	-	-
CO5	3	3	3	2	1	-	-	-	-	-	-	-
CO6	3	3	3	2	2	-	-	-	-	-	-	-
Avg	3.00	3.00	3.00	2.16	1.16	-	-	-	-	-	-	0.16

Highly Correlated: **3**

Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: Study the basics of cloud computing, cloud models and its applications.

CO2: Understand cloud services and architecture.

CO3: Learn how to use Cloud Services and to build applications.

CO4: Realize security needs for cloud service and Analyze different SLAs.

CO5: Analyze platform-specific security features and management of security controls.

CO6: Design, Develop & Deploy real-world applications in the cloud computing platforms.

OPERATION RESEARCH

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Operation Research	**COURSE CREDIT:** 02[2-0-0]
DEPARTMENT: Computer Science	**CATEGORY:** BS
CODE: XXXXXX	**SEMESTER:** 3^{rd}

THEORY

Learning objectives:

1) The course emphasizes 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) The students are expected to know and understand common and important engineering problems.
5) Students will develop problem modeling and solving skills and learn how to make intelligent decisions from the point of view of optimization.
6) The students will 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		15%
Module-III: Integer Programming Problems	6	15%
Module-IV: Goal Programming Problems	6	15%
Module-V: Markov Chains	6	15%
Module-VI: Game Theory	6	15%

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: [6L]
Formulation of Transportation Problems - Sensitivity Analysis in Transportation Problems - Assignment Problems

Module III: Integer Programming Problems: [6L]
Formulation, Cutting Plane Method - Branch and Bound Method – Applications

Module IV: Goal Programming Problems: [6L]
Single and Multiple Goal Programming Problems.

Module V: Markov Chains: [6L]
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.

- ***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.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: Apply operations research techniques like L.P.P, scheduling and sequencing in industrial optimization problems.

CO2: Solve allocation problems using various OR methods.

CO3: Analyze various OR models like Inventory, Replacement and apply them for optimization.

CO4: Analyze various OR models like Queuing, Decision and apply them for optimization.

CO5: Understand the concepts of integer linear programming.

CO6: Gain knowledge on current topics and advanced techniques of Operations Research in a wide range of applications in industries.

ACCOUNTING AND MANAGEMENT CONTROL

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Accounting and Management Control	**COURSE CREDIT:** 03[2-1-0]
DEPARTMENT: Computer Science	**CATEGORY:** HSM
CODE: XXXXXX	**SEMESTER:** 3^{rd}

THEORY

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	20%
Module-II: The Traditional Instruments of Control in Organizations	8	20%
Module-III: Accountability in Organizations	6	15%
Module-IV: The New Dimensions of Control with Strategies	6	15%
Module-V: ManagementControl inSpecializedOrganizations	4	15%
Module-VI: Non-profit organizations	4	15%

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.

- ***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)
2)

Reference Books:

1)

2)

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												
CO6												
Avg.												

Highly Correlated: **3**
Moderately Correlated: **2**
Slightly Correlated: **1**

Course Learning Outcome: (CO)

CO1: Apply management accounting and its objectives in facilitating decision-making and apply and analyze different types of activity-based management tools through the preparation of estimates.

CO2: Analyze cost-volume-profit techniques to determine optimal managerial decisions.

CO3: Perform cost variance analysis and demonstrate the use of standard costs in flexible budgeting.

CO4: Prepare analyses of various special decisions, using relevant management techniques.

CO5: Calculate various accounting ratios, reports and relevant data.

CO6: Prepare a master budget and demonstrate an understanding of the relationship between the components and prepare Cash Flow and Funds Flow statements this helps in planning for intermediate and long-term finances.

ARTIFICIAL INTELLIGENCE LAB

COURSE INFORMATION:

SCHOOL: School of Engineering	**COURSE TYPE:** L-T-P
NAME: Artificial Intelligence Lab	**COURSE CREDIT:** 02[0-0-2]
DEPARTMENT: Computer Science	**CATEGORY:** PC
CODE: XXXXXX	**SEMESTER:** 3^{rd}

List of practical

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	Type	Credit	Type		
					L	T	P
1	Major Project (6 months)		PSE	12	0	0	0
2	Grand Viva		PSE	6	0	0	0
3	Foreign Language – IV (German/ Spanish/ Japanese)		MUS	2	2	0	0
Total Credit				**20 Credit**			