KSR College of Engineering

NAAC ACCREDITED A++

NBA ACCREDITED PROGRAMMES

AN AUTONOMOUS INSTITUTION

MCA - MASTER OF COMPUTER APPLICATIONS

Curriculum & Syllabus for Semester I and II

REGULATIONS 2024

(Academic Year 2024-25 Onwards)

K.S.R. COLLEGE OF ENGINEERING: TIRUCHENGODE – 637215
(Autonomous)
DEPARTMENT OF COMPUTER APPLICATIONS
M.C.A. – Master of Computer Applications
(Regulations – 2024)

Vision of the institution

IV	We envision to achieve status as an excellent educational institution in the global knowledge hub, making self-learners, experts, ethical and responsible engineers, technologists, scientists, managers, administrators and entrepreneurs who will significantly contribute to research and environment friendly sustainable growth of the nation and the world.

Mission of the institution

IM 1	To inculcate in the students self-learning abilities that enable them to become competitive and considerate engineers, technologists, scientists, managers, administrators and entrepreneurs by diligently imparting the best of education, nurturing environmental and social needs.
IM 2	To foster and maintain a mutually beneficial partnership with global industries and institutions through knowledge sharing, collaborative research and innovation.

Vision of the Department/Programme: (Master of Computer Applications)

DV	To develop professionals having good knowledge, skills and attitude in the field of computer applications for the betterment of industry and society.

Mission of the Department/Programme: (Master of Computer Applications)

DM 1	To provide high quality education in the field of computer applications and there by create computer professionals with proper leadership skills, commitment and moral values.
DM 2	To educate students to be successful, ethical, and effective problem-solvers and life-long learners who will contribute positively to the economic well-being of our region and nation.

Program Educational Objectives (PEOs) : (Master of Computer Applications)

PEO 1	Demonstrate high quality fundamental knowledge in varied sectors and have the ability to develop innovative software on emerging technologies and provide access to higher degree by research programs.
PEO 1	Work as teams on multi-disciplinary projects with effective communication skills, critical thinking, individual, team work and leadership qualities necessary to function productively and professionally.
PEO 1	Understand the social and ethical professionalism, public policy and aesthetics that allows them to develop sufficient awareness of the societal impact of technology and the life-long learning needed for a successful professional career as a scientist / technocrat / an entrepreneur.

Program Outcomes (POs) : (Master of Computer Applications)

PO 1	**(Foundation Knowledge):** Apply knowledge of mathematics, programming logic and coding fundamentals for solution architecture and problem solving.
PO 2	**(Problem Analysis):** Identify, review, formulate and analyse problems for primarily focussing on customer requirements using critical thinking frameworks.
PO 3	**(Development of Solutions):** Design, develop and investigate problems with as an innovative approach for solutions incorporating ESG/SDG goals.
PO 4	**(Modern Tool Usage):** Select, adapt and apply modern computational tools such as development of algorithms with an understanding of the limitations including human biases.
PO 5	**(Individual and Teamwork):** Function and communicate effectively as an individual or a team leader in diverse and multidisciplinary groups. Use methodologies such as agile.
PO 6	**(Project Management and Finance):** Use the principles of project management such as scheduling, work breakdown structure and be conversant with the principles of Finance for profitable project management.
PO 7	**(Ethics):** Commit to professional ethics in managing software projects with financial aspects. Learn to use new technologies for cyber security and insulate customers from malware
PO 8	**(Life-long learning):** Change management skills and the ability to learn, keep up with contemporary technologies and ways of working.

	K.S.R. COLLEGE OF ENGINEERING (Autonomous) **(Approved by AICTE & Affiliated to Anna University)** **K.S.R. Kalvi Nagar, Tiruchengode - 637 215**	**CURRICULUM** **PG** **R - 2024**
Department	Computer Applications	
Programme	Master of Computer Applications	

SEMESTER - I

Sl.No.	Course Code	Course Name	Category	Hours/ Week			Credit	Maximum Marks		
				L	T	P	C	CA	ES	Total
THEORY										
01	MA24T14	Mathematical Foundation for Computer Applications	FC	3	0	0	3	40	60	100
02	CA24T11	Advanced Data Structures and Algorithms	PC	3	0	0	3	40	60	100
03	CA24T12	Computer Networks and Management	PC	3	0	2	4	40	60	100
04	CA24T13	Advanced Database Management Systems	PC	3	0	0	3	40	60	100
05	CA24T14	Python Programming	PC	3	0	0	3	40	60	100
06	CA24T15	Research Methodology and IPR	RM	3	0	0	3	40	60	100
PRACTICAL										
07	CA24P11	Advanced Data Structures and Algorithms Laboratory	PC	0	0	4	2	60	40	100
08	CA24P12	Advanced Database Management Systems Laboratory	PC	0	0	4	2	60	40	100
09	CA24P13	Python Programming Laboratory	PC	0	0	4	2	60	40	100
Total				**18**	**0**	**14**	**25**	**900**		

SEMESTER - II

Sl.No.	Course Code	Course Name	Category	Hours/ Week			Credit	Maximum Marks		
				L	T	P	C	CA	ES	Total
THEORY										
01	CA24T21	Advanced Java Programming	PC	3	0	0	3	40	60	100
02	CA24T22	Internet of Things	PC	3	0	0	3	40	60	100
03	CA24T23	Artificial Intelligence and Machine Learning	PC	3	1	0	4	40	60	100
04	CA24T24	Full Stack Development	PC	3	0	0	3	40	60	100
05		Professional Elective I	PE	3	0	0	3	40	60	100
06		Professional Elective II	PE	3	0	0	3	40	60	100
PRACTICAL										
07	CA24P21	Internet of Things Laboratory	PC	0	0	4	2	60	40	100
08	CA24P22	Advanced Java Programming Laboratory	PC	0	0	4	2	60	40	100
09	CA24P23	Full Stack Development Laboratory	PC	0	0	4	2	60	40	100
Total				**18**	**1**	**12**	**25**	**900**		

KSR College of Engineering	**K.S.R. COLLEGE OF ENGINEERING (Autonomous)** **(Approved by AICTE & Affiliated to Anna University)** **K.S.R. Kalvi Nagar, Tiruchengode - 637 215**	**CURRICULUM** **PG** **R - 2024**

Department	Computer Applications
Programme	Master of Computer Applications

SEMESTER – III

Sl.No.	Course Code	Course Name	Category	Hours/ Week			Credit	Maximum Marks		
				L	T	P	C	CA	ES	Total
THEORY										
01	CA24T31	Cloud Computing	PC	3	0	0	3	40	60	100
02	CA24T32	Big Data Analytics	PC	3	0	0	3	40	60	100
03	CA24T33	Software Testing & Quality Assurance	PC	3	0	0	3	40	60	100
04	CA24T34	Cryptography and Network Security	PC	3	0	0	3	40	60	100
05		Professional Elective III	PE	3	0	0	3	40	60	100
PRACTICAL										
06	CA24P31	Cloud Computing Laboratory	PC	0	0	4	2	60	40	100
07	CA24P32	Big Data Analytics Lab	PC	0	0	4	2	60	40	100
08	CA24P33	Mobile Application Development Laboratory	PC	0	0	4	2	60	40	100
Total				**1**	**0**	**12**	**21**	**800**		

SEMESTER – IV

Sl.No.	Course Code	Course Name	Category	Hours/ Week			Credit	Maximum Marks		
				L	T	P	C	CA	ES	Total
THEORY										
PRACTICAL										
01.	CA24P41	Project work	EE C	0	0	24	12	60	40	100
Total				**0**	**0**	**24**	**12**	**100**		

*** Total Credits: 83**

KSR College of Engineering	**K.S.R. COLLEGE OF ENGINEERING (Autonomous)** **(Approved by AICTE & Affiliated to Anna University)** **K.S.R. Kalvi Nagar, Tiruchengode - 637 215**	**CURRICULUM** **PG** **R - 2024**

Department	Computer Applications
Programme	Master of Computer Applications

List of Electives

ELECTIVE - I (SEMESTER - II)

Sl.No.	Course Code	Course Name	Category	Hours/ Week			Credit	Maximum Marks		
				L	T	P	C	CA	ES	Total
01	MA24E01	Probability And Statistics	PE	3	0	0	3	40	60	100
02	CA24E01	TCP/IP	PE	3	0	0	3	40	60	100
03	CA24E02	Unix and Network Programming	PE	3	0	0	3	40	60	100
04	CA24E03	Web Programming Essential	PE	3	0	0	3	40	60	100
05	CA24E04	Middleware Technology	PE	3	0	0	3	40	60	100
06	CA24E05	Devops	PE	3	0	0	3	40	60	100
07	BA24E51	Health Care Information Systems	PE	3	0	0	3	40	60	100

ELECTIVE - II (SEMESTER - II)

Sl.No.	Course Code	Course Name	Category	Hours/ Week			Credit	Maximum Marks		
				L	T	P	C	CA	ES	Total
01	MA24E02	Resource Management Techniques	PE	3	0	0	3	40	60	100
02	CA24E06	Software Project Management	PE	3	0	0	3	40	60	100
03	CA24E07	Advanced Operating Systems	PE	3	0	0	3	40	60	100
04	CA24E08	E-Learning Techniques	PE	3	0	0	3	40	60	100
05	CA24E09	Soft Computing	PE	3	0	0	3	40	60	100
06	CA24E10	Object Oriented Analysis and Design	PE	3	0	0	3	40	60	100
07	BA24E52	Organizational Behavior	PE	3	0	0	3	40	60	100

ELECTIVE - III (SEMESTER - III)

Sl.No.	Course Code	Course Name	Category	Hours/ Week			Credit	Maximum Marks		
				L	T	P	C	CA	ES	Total
01	CA24E11	M-Commerce	PE	3	0	0	3	40	60	100
02	CA24E12	Block chain Technology	PE	3	0	0	3	40	60	100
03	CA24E13	Robotic Process Automation	PE	3	0	0	3	40	60	100
04	CA24E14	Data Visualization Techniques	PE	3	0	0	3	40	60	100
05	CA24E15	Data Science	PE	3	0	0	3	40	60	100
06	CA24E16	Deep Learning	PE	3	0	0	3	40	60	100
07	CA24E17	Mobile Computing	PE	3	0	0	3	40	60	100

KSR College of Engineering	K.S.R. COLLEGE OF ENGINEERING (Autonomous) (Approved by AICTE & Affiliated to Anna University) K.S.R. Kalvi Nagar, Tiruchengode - 637 215	CURRICULUM PG R - 2024
Department	Computer Applications	
Programme	Master of Computer Applications	

BRIDGE COURSES

BRIDGE COURSES (SEMESTER - I)							
Sl.No.	Course Code	Course Name	Category	Hours/ Week			Credit
				L	T	P	C
01	BC5001	Computer Fundamentals	BC	2	0	0	0
02	BC5002	Problem Solving And Programming In C	BC	2	0	0	0
BRIDGE COURSES (SEMESTER -II)							
04	BC5003	Core Java Programming	BC	2	0	0	0
05	BC5004	Software Engineering	BC	2	0	0	0

Master of Computer Applications						
S.No.	Subject Area	Credit Per Semester				Credits Total
		Semester 1	Semester II	Semester III	Semester 4	
1	FC	3				3
2	PC	19	19	18		56
3	PEC		6	3		9
4	EEC				12	12
5	RMC	3				3
Total		25	25	21	12	83

FC – Foundation Course

PC – Professional Core Courses

PE – Professional Elective Courses

EEC – Employability Enhancement Courses

RMC -Methodology and IPR Courses

	K.S.R. COLLEGE OF ENGINEERING (Autonomous) **(Approved by AICTE & Affiliated to Anna University)** **K.S.R. Kalvi Nagar, Tiruchengode - 637 215**	**CURRICULUM** **PG** **R - 2024**
Department	Computer Applications	
Programme	Master of Computer Applications	

LIST OF VALUE ADDED COURSES/ONE CREDIT COURSES

Sl.No.	Subject Code	Course Name	Number of Hours	Offered by Internal / External
1	VACMCA01	Java Programming - Core Java	15 Hours	MCA/KSRCE
2	VACMCA02	Android Programming	15 Hours	MCA/KSRCE
3	VACMCA03	Advanced .Net Programming	15 Hours	MCA/KSRCE
4	VACMCA04	Python Programming Problem Solving	15 Hours	MCA/KSRCE
5	VACMCA05	Python Programming basics to Advanced	15 Hours	MCA/KSRCE
6	VACMCA06	Python Programming with R Programming	15 Hours	MCA/KSRCE
7	VACMCA07	Digital Marketing	15 Hours	External
8	VACMCA08	Course on Intellectual Property Rights	15 Hours	External
9	VACMCA09	General Physiology	15 Hours	External
10	VACMCA10	Waste Management	15 Hours	External
11	VACMCA11	Basic knowledge about Preening	15 Hours	External
12	VACMCA12	Yoga and Fitness Practical	15 Hours	External

MA24T14	**MATHEMATICAL FOUNDATION FOR COMPUTER APPLICATIONS**	**Category**	**L**	**T**	**P**	**C**
		FC	**3**	**0**	**0**	**3**

PREREQUISITE: The students should have basic knowledge in logical operations, set theory concepts, optimization problems, algebraic equations and manipulating expressions.

OBJECTIVES:

The Course will enable learners to:

- Describe, explain and apply the concepts of Logical connectives, Truth tables and inference theory.
- Apply the concepts of Statement functions, Logical equivalences.
- Construct and solve the problems involving linear programming models and apply simplex method to solve.
- Construct CPM and PERT to solve network problems.
- Apply various numerical methods to solve problems in algebraic, transcendental equation and interpolation.

UNIT - I PROPOSITIONAL CALCULUS (9)

Propositions – Logical connectives – Compound propositions - Conditional and biconditional propositions - Truth tables- Tautologies and contradictions - Logical equivalences and implications - Normal forms - Principal conjunctive and disjunctive normal forms - Rules of inference Theory.

UNIT - II PREDICATE CALCULUS (9)

Predicates - Statement functions – Variables - Free and bound variables – Quantifiers - Universe of discourse - Logical equivalences and implications for quantified statements - Theory of inference - The rules of universal specification and generalization.

UNIT - III LINEAR PROGRAMMING (9)

Formation of Linear Programming Problems - Graphical method - Simplex method – Big M Method - Dual simplex method.

UNIT - IV NETWORK ANALYSIS (9)

Network Construction - Critical Path Method - Computation of earliest start time, latest start time, Total, free and independent float time -PERT - Computation of optimistic, most likely Pessimistic and expected time – Probabilistic estimation for completion of project.

UNIT - V NUMERICAL TECHNIQUES (9)

Solution of algebraic and transcendental equation – Newton Raphson's method – Interpolation – Newton's forward and backward interpolation method- Lagrange's interpolation formula .

TOTAL: 45 PERIODS

COURSE OUTCOMES:

At the end of the course, the students will be able to:

Cos	***Course Outcomes***	**Cognitive Level (K1 to K6)**
CO1	Apply the propositional calculus.	Apply
CO2	Analyze the suitable predicate calculus	Analyze
CO3	Formulate and solve the Linear Programming problems.	Apply
CO4	Construct and solve the problems involving network analysis.	Apply
CO5	Compute and examine the roots of non-linear equations using numerical.	Analyze

TEXT BOOKS:

1. Tremblay J P, Manoharan R, "Discrete Mathematical Structures with Applications to Computer Science", McGraw-Hill, International Edition, 5th Edition, 2016.

2. Gupta P K, Man Mohan, "Operations Research", Sultan Chand & Sons Publications, New Delhi, 12th Edition, 2013

REFERENCES:

1. Bernard Kolman, Robert Busby, Sharon Cutler Ross, "Discrete Mathematical Structures for Computer Science", PHI Learning, 6th Edition, 2013.

2. Taha H A, “Operations Research”, Pearson Education , New Delhi, 9th Edition, 2012.
3. Grimaldi R P, Ramana BV,“Discrete and Combinatorial Mathematics”, Pearson Education, 5th Edition, 2006.
4. Kandasamy P, ThilagavathyK, Gunavathy K, “Numerical Methods”, S. Chand & Company Ltd, New Delhi,5th Edition, 2016.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	2	2	-	-	2
CO2	3	3	3	2	2	-	-	2
CO3	3	3	3	2	2	-	-	2
CO4	3	3	3	2	2	-	-	2
CO5	3	3	3	2	2	-	-	2
Avg.	3	3	3	2	2	-	-	2

1-low, 2-medium, 3-high

CA24T11	Advanced Data Structures and Algorithms	Category	L	T	P	C
		PC	3	0	0	3

Prerequisite:

Programming fundamentals: Understand the basics of programming, including variables, data types, loops, conditional statements, functions, and arrays. Discrete mathematics: Understand concepts like sets, relations, functions, and combinatory. Basic data types: Understand how to work with basic data types like integers, strings, and arrays. Problem-solving skills: Develop your ability to break down complex problems into smaller, manageable parts.

Course Objectives:

1. Understand the fundamentals: Learn the basic concepts, definitions, and terminology of data structures.
2. Data structure types: Understand the different types of data structures, including arrays, linked lists, stacks, queues, trees, graphs, and more.
3. Implementation and analysis: Learn how to implement and analyze various data structures, including their time and space complexity.
4. Problem-solving: Develop problem-solving skills using data structures to solve real-world problems.
5. Algorithm integration: Understand how to integrate data structures with algorithms to solve complex problems.

UNIT - I **INTRODUCTION** **(9)**

Introduction: Basic Terminology, Data type, Data object, Need of Data Structure, Types of Data Structure, Elementary Data Organization, Data Structure operations, Algorithm Complexity and Time-Space trade-off.

UNIT - II **ARRAYS** **(9)**

Arrays - Single and Multidimensional Arrays - address calculation - application of arrays - Character String in C - Sparse Matrices - Vectors - Searching: Sequential search - binary search - Sorting algorithms with efficiency - Bubble sort - Insertion sort - Merge sort - Quick Sort .

UNIT - III **STACKS** **(9)**

Stacks: Representation and Implementation of stack - Operations on Stacks: Push & Pop - Linked Representation of Stack - Operations Associated with Stacks. Applications of stack: Conversion of Infix to Prefix and Postfix Expressions - Evaluation of the postfix expression using stack. Recursion: Recursive definition and processes

UNIT - IV **QUEUES and TREES** **(9)**

Queues: Array and linked representation and implementation of queues - Operations on Queue: Create - Add - Delete - Full and Empty. Circular queue - DeQue, and Priority Queue. Linked list: Representation and Implementation of Singly Linked Lists. Trees: Basic terminology - Binary Trees - Binary tree representation - algebraic Expressions - Complete Binary Tree - Extended Binary Trees - BST - Traversing Binary trees

UNIT - V **GRAPH** **(9)**

Graph algorithms: Representations of graphs – Graph traversal: DFS – BFS – applications Connectivity, strong connectivity, bi-connectivity Minimum spanning tree: Kruskal's and Prim's algorithm- Shortest path: Bellman-Ford algorithm – Dijkstra's algorithm – Floyd- Warshall algorithm

Total 45 Periods

COURSE OUTCOMES*: At the edd of the course, the student will be able to:*

Cos	*Course Outcomes*	Cognitive Level (K1 to K6)
CO1:	Recognize the introduction about the Data Structures.	K2
CO2:	Summarize the fundamentals of Arrays Concept.	K2
CO3:	Discuss the concept of Stacks.	K4
CO4:	Discover the concept of Queues and Trees.	K4
CO5:	Utilize the concept of Graph.	K3

Reference Books:

1	A. M. Tanenbaum, Y. Langsam, M. J. Augustein, Data Structures using C, Prentice Hall of India, Delhi. Revised Edition 2023.
2	Yedidyah Langsam Moshe J. Augenstein, Aaron M. Tenenbaum, Data Structures using C & C++, PHI Publications, New Delhi, Second Edition 2017.
3	Mark Allen Weiss, Data structures and Algorithm Analysis in C++, Pearson Education. Ltd., Third Edition, 2016.
4	Michael T.Goodrich, R.Tamassia and D.Mount, Wiley, Data structures and Algorithms in C++, John Wiley

	and Son, Seventh Edition, 2016.
5	S.Sahan, Data structures Algorithms and Applications in C, Universities Press India Private Limited., Second Edition, 2015.
6	A. M. Tanenbaum, Data Structures using C, Prentice Hall of India, New Delhi., Revised Edition 2023.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	3	2	2	2	2
CO2	3	3	3	2	2	2	2	2
CO3	3	3	2	3	2	2	2	2
CO4	3	3	3	2	2	2	2	2
CO5	3	3	2	3	2	2	2	2
Avg.	**3**	**3**	**2.4**	**2.6**	**2**	2	2	**2**

1-low, 2-medium, 3-high

K.S.R. COLLEGE OF ENGINEERING (Autonomous)					**R 2024**
SEMESTER -I					
CA24T12	**Computer Networks and Management**	L	T	P	C
		3	0	0	3

Prerequisite:

Foundational Knowledge, Basic Computer Literacy, Understanding of Networking Fundamentals, Basic Knowledge of Computer Architecture, Familiarity with Network Devices, Basic Understanding of Network Security, Knowledge of Network Topologies and Architectures.

Course Objectives:

1.Understand Network Fundamentals
Protocols and Standards: Learn about essential networking protocols (TCP/IP, UDP, HTTP, FTP) and standards that govern network communication.
2. Design and Implement Network Architectures
Network Design: Understand the principles of network design, including scalability, reliability, and performance considerations. Learn to design network architectures that meet organizational needs.
3. Network Configuration and Administration
Network Management Tools: Use network management tools and software for monitoring, performance analysis, and troubleshooting.

UNIT - I DATA COMMUNICATION AND NETWORKING [09]

Data communication systems – Components and their functions - Building networks – Hosts and Networking devices – Switched Networks and Broadcast Networks – Transmission medium -Networking Devices: Hubs, Bridges, Switches, Routers, and Gateways -Role of software and hardware in networking – Layered Architecture – OSI and TCP/IP Reference Models.

UNIT - II PHYSICAL AND DATA LINK LAYERS [09]

Wired and wireless media – Functions of physical layer – Transmission errors – Error detection and correction – Framing - Flow control – Sublayers of DLL – Collision Domain - Ethernet – CSMA/CD – Token Ring – VLAN– IEEE 802.11 - WLAN – CSMA/CA – Bluetooth –
Ad hoc networks.

UNIT - III NETWORK LAYER [09]

Routing – Distance Vector and Link State Algorithms – RIP, OSPF and BGP -IPV4 Packet Format and Addressing – Effective IP address management techniques – CIDR – DHCP – ICMP – Need for IPv6 – Addressing methods and types in IPv6 – IPv6 header – Advantages of IPv6 – Transition from IPv4 to IPv6.

UNIT - IV TRANSPORT AND APPLICATION LAYERS [09]

Functions of transport layer - User Datagram Protocol – UDP Applications – Transmission Control Protocol – Connection establishment and release – Retransmission Strategies – Congestion Control – Application layer – Sockets – Protocols – HTTP – FTP- Email Protocols – DNS.

UNIT - V NETWORK MONITORING AND MANAGEMENT [09]

Network centric operations - Network monitoring – Open-source network monitoring tools – Network management model – Abstract Syntax Notation – Overview of MIB and SNMP – Wireshark tool – 16 Network provisioning – Fault detection, location and isolation.

LIST OF EXPERIMENTS

1. Practice different network commands available in Windows and Linux Operating Systems and troubleshoot the network.
2. Configure the network devices such as Router, Switch, Hub, Bridge and Repeater.
3. Analyzing the Network traffic using Packet Analyzer (Wireshark) and understanding the various protocol headers.
4. Configure IPv4 and IPv6 addressing for a network using static and dynamic approaches (SLAAC and DHCP).
5. Configure firewalls and honeypots
6. Performance analysis of Network using NS2/NS3/OPNET (Delay, Bandwidth etc.)

7. Develop client/server-based applications using TCP and UDP sockets.

Lecturer:45 Laboratory : 30 Total :75 Periods

COURSE OUTCOMES*: At the end of the course, the student will be able to:*

Cos	***Course Outcomes***	**Cognitive Level (K1 to K6)**
CO1:	*Explain Networking devices*	K1
CO2:	*Identify the Physical and Data Link Layers*	K2
CO3:	*Describe the Network Layer*	K3
CO4:	*Outline the Transport and application Layers*	K3
CO5:	*Illustrate Network Management*	K4

Reference Books:

1	Mani Subramanian, "Network Management: Principles and Practices", Second Edition, Pearson Education, 2015.
2	William Stallings, "Data and Computer Communications", Tenth Edition, Pearson Education, 2017.
3	Larry L. Peterson, Bruce S. Davie, "Computer Networks: A Systems Approach", Sixth Edition, Morgan Kaufmann Publishers Inc., 2022. 5.
4	Andrew S Tanenbaum, Nick Feamster and David J Wetherall, "Computer Networks", Sixth Edition, Pearson Education, 2022.
5	James F. Kurose, Keith W. Ross, "Computer Networking: A Top-Down Approach", Eighth Edition, Pearson Education, 2022.
6	Stephen Morris, "Network Management, MIBs and MPLS - Principles, Design and Implementation", Pearson Education, 2013.

Mapping of COs with POs and PSOs

COs/ POs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	3	2	3	2	3	2
CO2	3	3	3	3	3	2	3	2
CO3	3	3	3	2	3	2	2	3
CO4	3	3	3	3	3	2	3	3
CO5	3	3	3	2	3	2	3	3
Avg.	3	3	3	2	3	2	3	3

1-low, 2-medium, 3-high

CA24T13	Advanced Database Management Systems	Category	L	T	P	C
		FC	3	0	0	3

Prerequisite:

Programming Fundamentals: Knowledge of at least one programming language (like Python, Java, or C++) is often required, as advanced DBMS topics may involve writing code for database interactions, procedures, or applications. These concepts are often important when dealing with database optimization and management.

Course Objectives:

1. Understand the fundamentals: Master advanced SQL features like complex joins, sub queries, CTEs (Common Table Expressions), and window functions.
2. Transaction Management: Understand concepts like ACID properties (Atomicity, Consistency, Isolation, Durability), transaction isolation levels, and mechanisms for handling concurrent transactions.
3. Database Design: Learn about advanced data modeling techniques, such as normalization beyond the third normal form (3NF), denormalization, and multidimensional schema designs for OLAP.
4. Indexing and Optimization: Explore indexing strategies (e.g., B-trees, hash indexes), query optimization techniques, and the use of execution plans to improve performance.
5. Distributed Databases: Get familiar with the principles of distributing data across multiple locations, consistency models, data replication, and distributed query processing.

UNIT - I **INTRODUCTION** **[09]**

Database System Applications – Purpose of Database System. View of Data: Data Abstraction – Instances and Schemas – Data Models – Relational Database – Database Design –The Entity Relationship model.

UNIT - II **STORAGE AND FILE STRUCTURE** **[09]**

Overview of physical storage media – Magnetic Disks – Tertiary Storage – Storage Access. File Organization: Fixed Length Records – Variable Length Records. Organization of Records in Files: Sequential File Organization – Multi table Clustering File Organization – Data Dictionary Storage.

UNIT - III **RELATIONAL MODEL** **[09]**

Structure of Relational Databases –Fundamental Relational Algebra Operation. Transactions: Transaction Concept – Transaction State – Implementation of Atomicity and Durability – Concurrent Execution-Serializability.

UNIT - IV **SQL** **[09]**

Background – Data Definition- Basic Structure of SQL Queries – Set Operations – Aggregate Functions –Nested sub queries – Views – Joined Relations. Relational Database Design: Atomic Domain and First Normal Forms. Decom Position using Functional Dependencies: Keys and Functional Dependencies – Third Normal Form Boyce Codd Normal Form.

UNIT - V **INTRODUCTION OF PL/SQL** **[09]**

Advantages of PL/SQL –The Generic PL/ SQL Block. PL/SQL: Data types –Variables – Constants – Control Structures Cursors – Exception Handling –Procedures and Functions –Packages – Triggers.

Total 45 Periods

COURSE OUTCOMES*: At the edd of the course, the student will be able to:*

Cos	*Course Outcomes*	**Cognitive Level (K1 to K6)**
CO1:	Describe the need, role, importance and uses of databases.	K2
CO2:	Explain about storage and file structure.	K2
CO3:	Utilize the functions of Relational Model.	K4
CO4:	Write the query to perform the basic file operations.	K4
CO5:	Summarize the PL/SQL operations.	K3

Reference Books:

1	Abraham Silberschatz ,Henry F.Korth ,S.Sudarshan ,Database System Concepts, Tata McGraw Hill, Singapore, New Delhi, Fifth Edition, 2023.
2	Ivan Bayross, The Programming Languages of Oracle, BPB Publications, New Delhi, Third Edition, 2012.

3	C.J Date, An Introduction to Database System, Pearson Education, New Delhi, First Edition, 2015.
4	P.S.Deshpande, SQL & PL/SQL for Oracle 10g, Dream Tech Press, New Delhi, Third Edition, 2007.
5	Abraham Silberschatz, Hentry F.Korth and S.Sudharssan, Database System Concepts, Tata McGraw Hill, New Delhi , Fourth Edition, 2008.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	2	3	2	2	2
CO2	3	3	2	3	3	2	2	2
CO3	3	3	3	2	3	2	2	3
CO4	3	3	3	3	3	2	2	3
CO5	3	3	2	2	3	2	2	3
Avg.	3	3	2.4	2.4	3	2	2	2.6

1-low, 2-medium, 3-high

CA24T14	Python Programming	Category	L	T	P	C
		FC	3	0	0	3

Prerequisite:

Basic computer skills should be comfortable using a computer, typing and navigating through files and folders, Understanding of basic programming concepts like variables, data types, loops and control structures, Familiarity with a text editor or IDE: Need a text editor or Integrated Development Environment (IDE) to write and edit the Python code. Popular choices include Py Charm, Visual Studio Code, and Sublime Text.

Course Objectives:

1. Understand the basic syntax and data types in Python.
2. Learn control structures (if-else, for loops, while loops).
3. Understand string, lists and tuples.
4. Learn to work with dictionaries and functions.
5. Learn to handle files and modules.

UNIT - I INTRODUCTION TO PYTHON [09]

Introduction – Features – Downloading and Installing python - Executing a Python program – Flavors of Python – Memory Management in python – Garbage Collection – Comparisons between C and Python – Comparisons between Java and Python – Data types in Python: Comments –Built-in Data type – bool Data type – Sequences – Sets – literals – Identifiers and Reserved words – Naming Conventions.

UNIT - II LANGUAGE COMPONENTS [09]

Condition Statements: if , if-else statement. Looping Statement: While – for-Infinite loop – Nested loop - Break – Continue-Pass- Assert – Return. Operators – Input and Output Statements – Array: Creating – Importing the Array Module – Processing the Array – Types of Array – Operations on Arrays – Attributes of an Array- Case studies.

UNIT - III STRINGS, LISTS AND TUPLES [09]

Strings: Creating – Functions – Indexing - Slicing – Concatenation- Repetition – Membership – Comparing string – Testing Strings – Searching for sub strings – Converting Strings – Stripping White space Characters from a string – Formatting strings. Lists: Creating Lists – Updating -Concatenation - Repetition - Methods – Sorting- Nested Lists. Tuples: Creating - Accessing – Operations – Functions - Nested Tuples - Inserting Elements, Modifying Elements, Deleting Elements from a Tuple – Case studies.

UNIT - IV DICTIONARIES AND FUNCTIONS [09]

Dictionaries: Operations – Methods - Using for Loop with Dictionaries – Sorting the Elements of a Dictionary using Lambdas - Converting Lists and Strings into Dictionary - Passing Dictionaries to Functions - Ordered Dictionaries. Functions: Function Vs. Method - Defining – Calling – Returning - Pass by Object Reference – Arguments : Formal, Actual, Positional, Keyword, Default & Variable Length Arguments. Local and Global Variables - Recursive Functions - Lambdas - Function Decorators – Case studies.

UNIT - V FILES AND MODULES [09]

Files - Types of Files - Opening & Closing a File - Working with Text Files Containing Strings - Working with Binary Files - with Statement - seek() and tell() Methods - Random Accessing of Binary Files - Random Accessing of Binary Files using mmap - Zipping and Unzipping Files - Working with Directories. - Modules: Name spaces - Importing Modules – Module Built- in-functions- Standard Modules: math and dir function – Case studies.

Total 45 Periods

COURSE OUTCOMES*: At the end of the course, the student will be able to:*

Cos	***Course Outcomes***	**Cognitive Level (K1 to K6)**
CO1:	Explain basic principles of Python programming language.	K2
CO2:	Outline Language Components.	K2
CO3:	Utilize the Strings, Lists and Tuple Concepts.	K3
CO4:	Develop Dictionaries and Functions in Python.	K3
CO5:	Make use of Files and Modules.	K3

Reference Books:

1	Nageswara Rao, R., Core Python Programming, Dream tech Press, New Delhi, Second Edition,January 2018.
2	Robert Sedgewick, Kevin Wayne, Robert Dondero, Introduction to Programming in Python: An Inter-disciplinary Approach, Pearson India Education Services Pvt. Ltd., New Delhi, First Edition, 2016.
3	Daniel Liang Y., Introduction to Programming using Python, Pearson Education, New Delhi, Second Edition,
4	Wesley J. Chun, Core Python Programming, Pearson Education, New Delhi, Second Edition, 2010.
5	Kenneth A. Lambert, Fundamentals of Python: First Programs, Cengage Learning, New Delhi, 2016.
6	Guido van Rossum and Fred L. Drake Jr, An Introduction to Python Revised and updated for Python 3.2, Network Theory Ltd., New Delhi, 2011.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	2	3	2	2	2
CO2	3	2	3	2	3	2	2	2
CO3	3	3	3	2	3	2	2	2
CO4	3	2	3	2	3	2	2	2
CO5	3	3	2	2	3	2	2	2
Avg.	3	2.6	2.6	2	3	2	2	2

1-low, 2-medium, 3-high

CA24T15	Research Methodology and IPR	Category	L	T	P	C
		FC	3	0	0	3

Prerequisite: -

Research Methodology is a fundamental subject that equips students with the necessary skills to design, conduct, and evaluate research studies. It typically covers topics such as Research design and planning, Data collection and analysis methods, Statistical techniques, Research ethics. On the other hand, Intellectual Property Rights (IPR) deals with the legal aspects of protecting innovative ideas, creations, and inventions. IPR prerequisite may cover Copyright laws, Patent laws, Trademark laws, Trade secrets, Industrial design rights.

Course Objectives:

1. Understand the fundamentals of research design, methods, and techniques.
2. Develop skills to critically evaluate research studies and articles.
3. Understand the concept and importance of IPR.
4. Learn about different types of intellectual property (IP) rights (patents, copyrights, trademarks, etc.).
5. Understand the process of obtaining IP protection (filing, registration, etc.).

UNIT - I BASCIS OF RESEARCH PROBLEM (9)

Meaning of research problem – Sources of research problem – Criteria Characteristics of a good research problem – Errors in selecting a research problem – Scope and objectives of research problem. Approaches of investigation of solutions for research problem – Data collection – Analysis – Interpretation – Necessary instrumentations.

UNIT - II TECHNICAL WRITING AND PROPOSAL (9)

Effective literature studies approaches – Analysis Plagiarism – Research ethics – Effective technical writing – How to write Report – Paper – Developing Research Proposal – Format of research proposal – Presentation and Assessment by a review committee.

UNIT - III INTELLECTUAL PROPERTY (9)

Nature of Intellectual Property: Patents – Designs –Trade and Copyright. Process of Patenting and Development: Technological research – Innovation – Patenting – Development. International Scenario: International cooperation on Intellectual Property – Procedure for grants of patents – Patenting under PCT.

UNIT - IV PATENT RIGHTS (9)

Patent Rights: Scope of Patent Rights – Licensing and transfer of technology – Patent information and databases – Geographical Indications.

UNIT - V DEVELOPMENTS IN IPR (9)

New Developments in IPR: Administration of Patent System – New developments in IPR – IPR of Biological Systems – Computer Software – Traditional knowledge Case Studies – IPR and IITs..

Total (L= 45, T = 0) = 45 Periods

Course Outcomes : At end of the course, the student will be able to

COs	**Course Outcome**	**Cognitive Level (K1 to K6)**
CO1:	Recognize the introduction about research problem.	**K2**
CO2:	Analyze research related information.	**K4**
CO3:	Explain about Intellectual Property rights.	**K2**
CO4:	Utilize the Patent information and databases.	**K3**
CO5:	Emphasis the need of information about Intellectual Property Right	**K3**

Reference Books:

1	Ranjit Kumar, Research Methodology, A Step by Step Guide for beginners, SAGE Publications Asia Pacific Pvt Ltd, Singapore, Fourth Edition, 2023.
2	Robert P. Merges, Peter S. Menell, Mark A. Lemley, Intellectual Property in New Technological Age, Aspen Publishers, New York, And Sixth Edition 2016.
3	Stuart Melville and Wayne Goddard, Research methodology: an introduction for science & engineering students, Juta & Co , Kenwyn, South Africa, Second Edition,2014.

4	Wayne Goddard and Stuart Melville, Research Methodology, An Introduction, Juta and Company Ltd, Lansdowne, Second Edition, 2014.
5	Stuart Melville and Wayne Goddard, Research methodology: an introduction for science & engineering students, Lansdowne, Juta and Company Ltd, Second Edition, 2014.
6	T. Ramappa, Intellectual Property Rights Under WTO, S. Chand, Wheeler Publishing, Hyderabad, Second Revised Edition, 2016
7	Halbert, Resisting Intellectual Property, Taylor & Francis Ltd, London, second Edition, 2012.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	2	3	3	2	2	3
CO2	3	2	2	3	3	2	2	3
CO3	3	3	3	3	3	3	2	3
CO4	3	3	3	3	3	3	2	3
CO5	3	3	3	3	3	3	3	3
Avg.	3	2.6	2.6	3	3	3	2.2	3

1-low, 2-medium, 3-high

CA24P11	Advanced Data Structures and Algorithms Laboratory	Category	L	T	P	C
		FC	0	0	4	2

Prerequisite:

Programming fundamentals: Students should have a solid grasp of programming concepts, including data types, variables, control structures, functions, and arrays. Introduction to programming: Students should have completed an introductory programming course, such as Python, Java, C++, or JavaScript. Algorithms basics: Students should have a basic understanding of algorithms, including sorting, searching, and graph traversal.

Course Objectives :

1. Hands-on implementation: Implement various data structures, such as arrays, linked lists, stacks, queues, trees, and graphs, using a programming language.
2. Experimentation and analysis: Experiment with different data structures, analyze their performance, and compare their trade-offs.
3. Problem-solving: Apply data structures to solve real-world problems and laboratory exercises.
4. Debugging and testing: Develop debugging and testing skills to ensure the correctness of implemented data structures.
5. Code optimization: Learn to optimize code for efficiency, readability, and scalability.

LIST OF EXPERIMENTS

1. Write a C program for implementation of stack using array
2. Write a C program for implementation of queue using array
3. Write a C program for implementation of circular queue using array
4. Write a C program evaluate postfix expression using stack.
5. Design, develop and execute a program in C to read a sparse matrix of integer values and make a transpose of it. Use the triple to represent an element in sparse matrix.
6. Design, develop and execute a program in C to implement singly linked list where each node consist of integers.
7. Write a C program evaluate sort a given set of elements using Quick sort.
8. Write a C program evaluate sort a given set of elements using Merge sort.
9. Compute the transitive closure of a given directed graph using Warshall’s algorithm.
10. Find minimum cost spanning tree for a given undirected graphs using Kruskls algorithms

Total 30 Periods

Course Outcomes : At the end of the course, the student will be able to:

COs	Course Outcome	Cognitive Level (K1 to K4)
CO1:	Develop the programming skills in design and implementation of data structures and their applications	K4
CO2:	Write and execute programs in C to implement circular queue and evaluation of postfix expression	K4
CO3:	Write and execute programs in C to implement of sparse matrix and singly linked lsit	K4

CO4:	Develop and execute programs for sorting techniques.	K6
CO5:	Create the C Program to find transitive closure and minimum spanning tree of a given graph.	K6

REFERENCES :

1. Reema Thareja,Data Structures using C, Oxford University Press, Third Edition, 2023.
2. Hemant Jain, Problem Solving in Data Structures & Algorithms Using C, Third Edition, 2022.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	3	2	2	2	2
CO2	3	3	3	2	2	2	2	2
CO3	3	3	2	3	2	2	2	2
CO4	3	3	3	2	2	2	2	2
CO5	3	3	2	3	2	2	2	2
Avg.	3	3	2.4	2.6	2	2	2	2

1-low, 2-medium, 3-high

CA24P12	Advanced Database Management Systems Laboratory	Category	L	T	P	C
		FC	0	0	4	2

Prerequisite:

Programming fundamentals: Familiarity with fundamental database design and SQL skills. Intermediate SQL: Proficiency in writing and optimizing more complex SQL queries and scripts. Database Design Skills: Understanding of data modeling, including ER diagrams and normalization. Programming Knowledge: Experience with programming languages used for database interactions, such as Python, Java, or C++. Introductory DBMS Laboratory: Completion of a basic database laboratory course or equivalent hands-on experience with database tools and environments.

Course Objectives :

1. Hands-on implementation: Implement transactions with rollback, commit, and save point operations. Handle concurrency and isolation levels.
2. Experimentation and analysis: Experiment with different data structures, analyze their performance, and compare their trade-offs.
3. Problem-solving: Design a database schema for a given complex application or business scenario.
4. Debugging and testing: Check database and application logs for error messages or warnings that can provide clues.
5. Code optimization: Replace sub queries with joins where possible, as joins are often more efficient.

LIST OF EXPERIMENTS

1. Design a Table and Execute DDL, DML and DCL Queries.
2. Design a Table and Execute Aggregate Functions and Set Operations.
3. Write a Program on Normalization.
4. Execute a SQL Command to Perform Sub Queries and Joins.
5. Implement a Nested Sub Queries and Correlated Sub Queries.
6. Create Views for a Particular Database.
7. Implement a PL/SQL Procedure for an Application Using Procedure.
8. Implement a PL/SQL Procedure for an Application Using Function.
9. Implement a SQL Comments Triggers.
10. Implement a PL/SQL Procedure for an Application Using Exception Handling.

Total 30 Periods

Course Outcomes : At the end of the course, the student will be able to:

COs	Course Outcome	Cognitive Level (K1 to K4)
CO1:	Explain basic database concepts, applications, data models, schemas and instances.	K4
CO2:	Demonstrate the use of constraints and relational algebra operations.	K4

CO3:	Emphasize the importance of normalization in databases.	K4
CO4:	Describe the basics of SQL and construct queries using SQL.	K6
CO5:	Explain the familiarize issues of concurrency control and transaction management.	K6

REFERENCES :

1. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, "Database System Concepts", 6th edition, Tata McGraw Hill, 2011
2. Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", 4th Edition, Pearson/Addision wesley, 2007

Outcome:

1. Ability to use databases for building web applications.
2. Gaining knowledge about the internals of a database system.

Mapping of COs with POs and PSOs

COs/ POs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	2	3	2	2	2	2
CO2	3	3	3	2	2	2	2	2
CO3	3	3	2	3	2	2	2	2
CO4	3	3	3	2	2	2	2	2
CO5	3	3	2	3	2	2	2	2
Avg.	3	3	2.4	2.6	2	2	2	2

1-low, 2-medium, 3-high

CA24P13	Python Programming Laboratory	Categor y	L	T	P	C
		FC	0	0	4	2

Prerequisite:

Foundational Knowledge

Basic Computer Literacy

Mathematical Skills

Introduction to Programming Concepts

Basic Programming Knowledge

Course Objectives :

1.Understand Python Fundamentals
2.Develop Problem-Solving Skills
3.Master Data Structures and Algorithms
4.Work with Libraries and Modules
5.Develop and Debug Python Code

LIST OF EXPERIMENTS

1. Program using Operators
2. Program using Conditional Statements
3. Program using Looping
4. Program using Strings
5. Program using Lists
6. Program using Dictionaries
7. Program using Tuples
8. Program using Functions
9. Program using File handling
10. Program using Modules
11. Develop the simple project

Total 30 Periods

Course Outcomes : At the end of the course, the student will be able to:

COs	Course Outcome	Cognitive Level (K1 to K4)
CO1:	Describe the Python language syntax including control statements,	Understand
CO2:	Write programs for a wide variety problem in mathematics,	Apply
CO3:	Write Test and Debug Python Programs	Apply
CO4:	Implement Conditionals and Loops for Python Programs..	Apply
CO5:	Illustrate the Use functions and represent Compound data using	Develop

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	3	2	2	2	2
CO2	3	3	3	2	2	2	2	2
CO3	3	3	2	3	2	2	2	2
CO4	3	3	3	2	2	2	2	2
CO5	3	3	2	3	2	2	2	2

Avg.	**3**	**3**	**2**	**3**	**2**	**2**	**2**	**2**
1-low, 2-medium, 3-high								

CA24T21	**Advanced Java Programming**	**Category**	L	T	P	C
		FC	3	0	0	3

PREREQUISITE

Introduction to programming: Students should have a basic understanding of programming concepts, including data types, variables, control structures, functions, and arrays. Familiarity with programming fundamentals: Students should have experience with programming concepts, such as loops, conditional statements, and functions. Basic understanding of object-oriented programming (OOP) concepts: Students should have a basic understanding of OOP concepts, including classes, objects, inheritance, polymorphism, and encapsulation.

OBJECTIVES:

1.Understand Java fundamentals: Learn the basic syntax, data types, variables, operators, control structures, functions, and object-oriented programming (OOP) concepts in Java.

2. Write Java programs: Develop skills to write efficient, readable, and well-documented Java programs using various data types, control structures, and OOP concepts.

3. Understand Java libraries and frameworks: Familiarize yourself with Java libraries and frameworks, such as Java Standard Library, Java Collections Framework, and Java Stream API.

4. Develop problem-solving skills: Apply Java programming skills to solve real-world problems, including debugging and testing.

5. Understand Java best practices: Learn Java coding standards, naming conventions, and best practices for writing efficient and maintainable code.

UNIT - I	**OVERVIEW OFJAVA**	**(9)**

Introduction-Java-Object Oriented Programming Concepts- Data Types- Variables and Arrays – Control Statements–Method Overriding.

UNIT - II	**PACKAGES**	**(9)**

Packages – Importing Packages- I/O Package – Interfaces – Exception Handling – Multithreaded Programming- String Operations. The Java I/O Classes– File – Byte Streams – The Character Streams – Serialization. AWT Package: AWT Classes – Window Fundamentals – Working with Graphics– Working with Color – Working withFonts – Applet Package: Applet Basics – Applet Architecture – Reading and Writing in Console – Print Writer class.

UNIT - III	**NETWORK PROGRAMMING IN JAVA**	**(9)**

Sockets – secure sockets – custom sockets – UDP data grams – multicast sockets –URL classes – Reading Data from the server – writing data – configuring the connection – Reading the header – telnet application – Java Messaging services.

UNIT - IV	**APPLICATIONS IN DISTRIBUTED ENVIRONMENT**	**(9)**

Remote method Invocation – activation models – RMI custom sockets – Object Serialization – RMI – IIOP implementation – CORBA – IDL technology – Naming Services – CORBA programming Models - JAR file creation

UNIT - V	**EVENT-DRIVEN PROGRAMMING**	**(9)**

Graphics programming – Frame – Components – working with 2D shapes – Using color, fonts, and images - Basics of event handling – event handlers – adapter classes – actions – mouse events – AWT event hierarchy – introduction to Swing – Model-View- Controller design pattern – buttons – layout management – Swing Components.

Total 45 Periods

Course Outcomes : At end of the course, the student will be able to

COs	**Course Outcome**	**Cognitive Level** (K1 to K4)
CO1:	Explain the basic programming concepts of java.	K1
CO2:	Defined package, to create thread program and string	K2
CO3:	Examine the input/output and networking package	K3
CO4:	Explore the abstract Applications in Distributed	K1
CO5:	Illustrate the Even – Driven Programming.	K3

Reference Books :

1	Herbert Schildt, The Complete Reference JAVA, Tata McGraw Hill , Noida, Thirteenth Edition , 2017.
2	Gavin King, Java Persistence with Hibernate, Manning Publications, United States, Second Edition, 2016
3	Anuradha A. Puntambekar, Advance Java,Technical Publications, First Edition, New Delhi, 2021.
4	Kogent, Java 6 Programming Black Book, Kogent Learning Solutions, New Delhi, 2017.
5	Steven Holzner, Java2 (JDK 5 Edition) Programming, Dreamtech Press India Pvt. Ltd, New Delhi, 2016
6	Prem Kumar, Getting Inside Java - Beginners Guide, Pencil, First Edition, 2021.

Mapping of COs with POs and PSOs

COs/ POs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	2	3	2	2	2	2
CO2	3	3	3	2	2	2	2	2
CO3	3	3	2	3	2	2	2	2
CO4	3	3	3	2	2	2	2	2
CO5	3	3	2	3	2	2	2	2
Avg.	3	3	2.4	2.6	2	2	2	2

1-low, 2-medium, 3-high

CA24T22	Internet of Things	Category	L	T	P	C
		FC	3	0	0	3

Prerequisite:

1. Connectivity: Devices must be able to connect to the internet or a local network.
2. Sensors and Actuators: Devices need sensors to collect data and actuators to perform actions.
3. Microcontrollers or Processors: Devices need a brain to process data and control actions.
4. Power Supply: Devices need a reliable power source, such as batteries or electricity.
5. Communication Protocols: Standardized protocols (e.g., Wi-Fi, Bluetooth, MQTT) enable devices to communicate.
6. Data Storage and Analytics: Cloud or local storage and analytics capabilities to process and make sense of data.
7. Security: Measures to ensure secure data transmission, storage, and device protection.
8. Interoperability: Devices and systems must be able to communicate and work together seamlessly.

Course Objectives:

1. Understand IoT fundamentals
2. Learn IoT architecture
3. Develop device programming skills
4. Understand sensor and actuator integration
5. Study communication protocols

UNIT - I **TECHNOLOGIES IN INTERNET OF THINGS** **[09]**

Various Technologies Used in IoT - IoT Revolution - Benefits of IoT - IoT Frameworks: Value Chain - Frameworks and Platforms -AWS IoT– Waston IoT Platform-IoT Ecosystem - Elements for IoT Implementation - Case Studies: E-Health System – Environmental Monitoring.

UNIT - II **COMPONENTS IN INTERNET OF THINGS: DESIGN PATTERN, IOT ARCHITECTURE AND CORE MODULES** **[09]**

Various Design Patterns - Challenges and Solutions for Designing Architecture for IoT - Four Layer Architecture - Seven Layer Architecture - Core Modules: Protocols – Sensors – Endpoints - Data Communication - IoT Data Managements - Analytics.

UNIT - III **IOT IMPLEMENTATION** **[09]**

IoT Implementation Strategies: Challenges and Solutions - Things to Know Before an IoT Implementation - Types of Testing in IoT- Testing Challenges and Tools - Testing Smart Wearables. Case Studies: Monitoring Traffic Volume in Petrol Stations to Improve Sales Strategies, Smartphone Detection System in the Crowd.

UNIT - IV **TECHNOLOGIES BEHIND IOT** **[09]**

Artificial Intelligence for IoT: Exploring the world of AI, IoT and AI in the context of Industry 4.0 – Data Analytics and Machine Learning for IoT – Security Challenges for IoT.

UNIT - V **INTERNET OF THINGS IN INDUSTRY** **[09]**

Industries: Manufacturing – Oil and Gas – Transportation – Public Safet.

Total 45 Periods

COURSE OUTCOMES: At the end of the course, the student will be able to:

Cos	**Course Outcomes**	**Cognitive Level (K1 to K6)**
CO1:	Explain Working Principles of Different Technologies With IOT Platforms.	K1
CO2:	Describe About the Components of IOT, IOT Architecture and Core Modules.	K2
CO3:	Demonstrate The Process of IOT Implementation in Various Applications.	K3
CO4:	Explain the Various Technologies Behind IOT and Industrial IOT	K3
CO5:	Implementation of IOT Using Different Sensors to Solve the Real World Problems.	K3

Reference Books:	
1	David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry “IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things”, 9th Impression, Pearson Publication, 2022. (UnitV)
2	MayurRamgir, “Internet of Things- Architecture, Implementation, and Security”, 1st Edition, Pearson Publication, 2020. (Unit I - IV)
3	Dr. OvidiuVermesan and Dr. Peter Friess, “Internet of Things: From research and innovation to market deployment”, River Publishers, 2014.

Mapping of COs with POs and PSOs

COs/ POs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	2	3	3	3	3	1	2
CO2	3	3	3	3	2	3	1	3
CO3	2	3	3	3	2	3	1	2
CO4	3	2	3	3	2	3	2	3
CO5	3	3	3	2	2	2	2	3
Avg.	2.8	2.6	3	2.8	2.2	2.6	1.6	2.6

1-low, 2-medium, 3-high

CA24T23	Artificial Intelligence and Machine Learning	Category	L	T	P	C
		FC	3	1	0	4

Prerequisite:

Understanding vectors, matrices, and operations on these is key, as many ML algorithms rely on linear algebra concepts. Differentiation and integration are important for understanding optimization techniques used in training models. Knowledge of probability distributions, statistical inference, and data analysis methods is essential for designing and evaluating ML models. Python is the most widely used language in AI and ML due to its extensive libraries. Familiarity with variables, loops, conditionals, and data structures is important.

Course Objectives:

1. Understanding Core Concepts : Understanding of the basic principles of AI and ML, including the supervised, unsupervised, and reinforcement learning.
2. Mathematical and Statistical Foundations :Develop proficiency in linear algebra, calculus, probability, and statistics as they apply to AI and ML algorithms.
3. Practical Application and Implementation: Build and refine skills in programming, particularly in Python, and use relevant libraries
4. Data Handling and Preparation :Learn techniques for cleaning, transforming, and preparing data for analysis and modeling.
5. Model Evaluation and Improvement: Learn how to use metrics such as accuracy, precision, recall, F1 score,

UNIT - I	INTRODUCTION TO AI SEARCH AND OPTIMIZATION ALGORITHM	(9)

The Foundations of Artificial Intelligence - History of AI - State of the Art -how do AI algorithm work-Types of AI:Supervised Learning-Unsupervised Learning-Reforcement Learning- Creating your own Chatbot using DialogFlow Introduction to search algorithm-Greedy search algorithm-Hill Climbing Algorithm- A* Search algorithm

UNIT - II	FUZZY LOGIC	(9)

Introduction to Fuzzy Logic-Crisp sets-Fuzzy Sets-Architecture of Fuzzy Logic-Membership Function-Fuzzy Logic Algorithm-Fuzzy control-Sugeno style of Fuzzy Inference Processing-Fuzzy Hedges-alpha cut Treshold-Neuro Fuzzy systems.

UNIT - III	MACHINE LEARNING	(9)

Classification-Navie Bayes-Decision Tree-Rule based learning-k nearest neighbour -Suppport vector machine-Regression-Linear Regresion-Logistic Regression-Multivariate Regression-Multiple Regression-Clustering-K-means Clustring-Fuzzy C-means-EM algorithm-Hierarchical clustering algorithm.Machine learning Application: Heart Disease Prediction- Titanic Survival using NAÏVE BAYES.

UNIT - IV	NEURAL NETWORKS AND DEEP LEARNING	(9)

Neural Networks

Introduction to Artificial Neural Networks-Artificial neuron Model and Linear Regression-Gradient Descent Algorithm-Learning Mechanism-Hebbian, Competitive, Boltzmann-Associative memory.

Deep Learning

Introduction – Neurons and Activation Functions-Deep learning libraries- Image Classification using Convolutional Neural Networks (CNN) -Hand Gesture Recognition with Deep Learning-Detecting Plant Leaf Diseases using Deep Learning- Drowsiness Detection for Driver Safety

UNIT - V	NLP AND COMPUTER VISION	(9)

Introduction to NLP-Types of NLP algorithm: Sentiment analysis, Keyword extraction, Knowledge graphs, Wordclouds,Text summarization - Applications:Generating Titles from Text using NLP- Analyzing Speech and Emotions using CNN and NLP Introduction to coputer vision-history of Computer vision-Task associated with computer vision -applications:Moving Object Detection using OpenCV Advanced Face Detection and Tracking.

Total 45 Periods

COURSE OUTCOMES: At the end of the course, the student will be able to:

Cos	Course Outcomes	Cognitive Level (K1 to K6)
CO1:	Illustrate the principles and approaches with real time problems	K2
CO2:	Learn various techniques of artificial intelligence building blocks and their applications.	K4
CO3:	Understand knowledge representation methodologies and principles of reasoning.	K2
CO4:	Analyze fuzzy logic systems; various neural network architecture and their applications.	K3
CO5:	Solve problems in various applications using genetic algorithms and artificial immune system.	K2

Reference Books:

1	Nigel Toon, How AI Thinks: How we built it, how it can help us, and how we can control it Paperback ,8 February 2024
2	VaibhavVerdhan ,Computer Vision Using Deep Learning Neural Network Architectures with Python and Keras Paperback Import, 15 February 2021
3	Laurence Moroney , AI and Machine Learning for Coders A Programmer's Guide to Artificial Intelligence , Grayscale Indian Edition Paperback – 25 October 2020
4	Eduonix Learning Solutions, Machine Learning for Healthcare Analytics Projects Paperback – Large Print, 29 October 2018
5	Ajit K Jha , A Strategic Approach to Artificial Intelligence Projects: Learn AI & Ml Without Coding, AI Project Lifecycle, AI Leadership Approach, Ai In Cyber security, Intelligence for managers and leaders, Kindle Edition.
6	kevinKnight,ElaineRich,B.Nair , Artificial Intelligence, , Paperback, Third Edition, July 2017.
7	RajendraAkerkar , Introduction To Artificial Intelligence ,PHI Learning,2nd Edition, July 2014.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	3	3	2	2
CO2	3	2	3	3	2	3	1	3
CO3	2	3	2	3	2	3	2	2
CO4	3	2	3	3	2	3	2	3
CO5	3	3	3	2	2	2	3	3
Avg.	2.6	2.4	2.8	2.8	2.2	2.8	2	2.6

1-low, 2-medium, 3-high

CA24T24	Full Stack Development	Category	L	T	P	C
		FC	3	0	0	3

Prerequisite:

Full stack development involves working on both the front-end and back-end aspects of web applications. Experience with JavaScript libraries and frameworks such as React, Angular, or Vue.js for building dynamic user interfaces. Understanding of CI/CD pipelines and tools like Jenkins, GitLab CI, CircleCI, or Travis CI. Proficiency in using Integrated Development Environments (IDEs) or code editors like VSCode, Sublime Text, or Atom.

Course Objectives:

1. Understand the working on both the front-end and back-end of web applications,
2. Understand the client-side actions.
3. Utilize modern frameworks, libraries, and tools to streamline development processes and leverage current technology trends.
4. Learn about basic concepts of Web pack.
5. Understanding how to create and manage Docker containers. ROC-AUC, and confusion matrices to assess model performance

UNIT - I	SERVER-SIDE ACTION	(9)

Node and NPM - Installation - Commands - Packaging – file system - http/ https - OS - Path - Process - collaborative version control system git – Introduction to MERN stack.

UNIT - II	CLIENT-SIDE ACTIONS	(9)

React - Writing different components - Introduction to Typescript - Programming structures - Boolean - Arrays - Tuples - function.

UNIT - III	ADVANCED TYPESCRIPT	(9)

Classes - Inheritance - Interfaces - Namespaces - Modules - Decorators - Debugging Typescript apps - development of a simple web application with typescript.

UNIT - IV	WEBPACK	(9)

Introduction to web pack - dependency graph - Plugins - Modules - Adding node modules - REST Endpoint - mailer - other examples

UNIT - V	DEPLOYMENT THROUGH CONTAINERS	(9)

Containerization - Installation of Docker - Pulling Images - Creating Images - Deploying to Dockerhub - Development and deployment of js applications in docker.

Total 45 Periods

COURSE OUTCOMES: At the end of the course, the student will be able to:

Cos	Course Outcomes	Cognitive Level (K1 to K6)
CO1:	Work with collaborative version control	K3
CO2:	Develop web applications using jsNode	K3
CO3:	Use Typescript for Client-side actions	K4
CO4:	Explore web pack for creating web applications	K4
CO5:	Develop web applications with Typescript. Deploy Web applications through containers	K3

Reference Books:

1	Vasan Subramanian, Pro MERN Stack, Full Stack Web App Development with Mongo, Express, React, and Node, A Press Publisher, Second Edition 2019.
2	Karl Seguin, The Little Mongo DB Book, O Reilly, First Edition, 2018.
3	Gareth Dwyer, Flask by Example, Packt Publishers, Second Edition, 2016.

4	Gene Kim, Kevin Behr, George Spafford, The Phoenix Project, A Novel about IT, DevOps, IT Revolution Press, Fifth Edition, 2018.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	3	3	2	2
CO2	3	2	3	3	2	3	1	3
CO3	2	3	2	3	2	3	2	2
CO4	3	2	3	3	2	3	2	3
CO5	3	3	3	2	2	2	3	3
Avg.	3	3	3	3	2	3	2	3

1-low, 2-medium, 3-high

CA24P21	Internet of Things Laboratory	Category	L	T	P	C
		FC	0	0	4	2

Prerequisite:

1. Microcontrollers (Arduino, Raspberry Pi, ESP32)
2. Sensors (temperature, humidity, motion, light)
3. Actuators (LEDs, motors, relays)
4. Development boards (breadboards, PCBs)
5. Power supplies (batteries, wall adapters)
6. Communication modules (Wi-Fi, Bluetooth, Ethernet)

Course Objectives :

1. Research and Development.
2. Technology Integration
3. Application Development
4. Data Management and Analytics
5. Security and Privacy

LIST OF EXPERIMENTS

1. Familiarization with concept of IoT, Arduino / Raspberry-Pi and perform necessary software installation.
2. Study of different operating systems for Raspberry-Pi. Understanding the process of OS installation on Raspberry-Pi.
3. Study of connectivity and configuration of Raspberry-Pi with basic peripherals, LED ON / OFF using Push Button, understanding GPIO and its use in program.
4. Understanding and connectivity of Raspberry-Pi with Distance measuring using Ultrasonic Sensor. Write an application to measure the distance of the obstacle using Ultrasonic Sensor.
5. Understanding and connectivity of Raspberry-Pi with Temperature and Humidity Sensor. Write an application to read an environment temperature and Humidity value. If a temperature crosses a threshold value, the application indicated user using LEDs.
6. Understanding and connectivity of Raspberry-Pi with IR Sensor. Write an application to detect obstacle and notify user using LEDs.
7. Understanding and connectivity of Raspberry-Pi with camera. Write an application to detect the color of the object or obstruction detection.
8. Write an application using Raspberry-Pi based health monitoring using heartbeat and Pulse Sensor
9. Write an application using Raspberry-Pi based Eye blinking/closeness detection sensor.
10. Write an application using Raspberry-Pi based Rain fall detection using Rain Sensor.

Total 30 Periods

Course Outcomes : At the end of the course, the student will be able to:

COs	Course Outcome	Cognitive Level (K1 to K4)
CO1:	Understanding IoT Fundamentals and the process of OS installation on Raspberry-Pi. Raspberry-Pi.	Understand
CO2:	Device and Sensor Integration	*Apply*
CO3:	Data Collection and Analysis	*Apply*

CO4:	IoT Protocols and Communication.	Apply
CO5:	Programming and Development Skills	*Develop*

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	3	3	2	2
CO2	3	2	3	3	2	3	1	3
CO3	2	3	2	3	2	3	2	2
CO4	3	2	3	3	2	3	2	3
CO5	3	3	3	2	2	2	3	3
Avg.	3	3	3	3	2	3	2	3

1-low, 2-medium, 3-high

CA24P22	Advanced Java Programming Laboratory	Category	L	T	P	C
		FC	0	0	4	2

PREREQUISITE

Introduction to Programming: Students should have a basic understanding of programming concepts, including data types, variables, control structures, functions, and arrays. Programming in Java: Students should have completed a course in Programming in Java or have equivalent experience with the Java programming language. Object-Oriented Programming (OOP) concepts: Students should have a basic understanding of OOP concepts, including classes, objects, inheritance, polymorphism, and encapsulation.

OBJECTIVES:

1. Apply Java programming skills: Students will apply their Java programming skills to solve real-world problems and complete laboratory exercises.

2. Develop problem-solving skills: Students will develop problem-solving skills using Java programming, including debugging and testing.

3. Understand Java libraries and frameworks: Students will learn to utilize Java libraries and frameworks to develop efficient and effective programs.

4. Improve coding skills: Students will improve their coding skills, including writing efficient, readable, and well-documented code.

5. Collaborate with peers: Students will collaborate with peers to complete laboratory exercises and projects.

LIST OF EXPERIMENTS

1. Write a Java Program to Illustrate the use of Object Oriented Programming Concepts

2. Write a java Program to represent Array List class.

3. Write a Java Program to Illustrate the use of Overriding.

4. Write a Java Program to Implement String Handling Functions.

5. Write a Java Program to Implement any 4 File Operations.

6. Create a Calculator Using AWT Controls and use Event Handling for Calculations.

7. Write a java program to Implement Action Listener.

8. Create a Java Application using Packages.

9. Create a Java Application using RMI

10. To Develop an Applet Program using Sockets.

Total : 30 Periods

COURSE OUTCOMES: At the end of the course, the students will be able to:

COs	Course Outcome	Cognitive Level (K1 to K4)
CO1	Demonstrate the concepts of Object Oriented Programming.	*K1*
CO2	Implement the concepts of overriding.	*K1*
CO3	Perform the concept of Handling.	*K4*
CO4	Develop a Program using Packages.	*K4*
CO5	Perform the program using Applet.	*K4*

REFERENCES:

1. Herbert Schildt, The Complete Reference JAVA, Tata McGraw Hill , Noida, Thirteenth Edition , 2017.
2. Gavin King, Java Persistence with Hibernate, Manning Publications, United States, Second Edition, 2016.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	3	3	2	2
CO2	3	2	3	3	2	3	1	3
CO3	2	3	2	3	2	3	2	2
CO4	3	2	3	3	2	3	2	3
CO5	3	3	3	2	2	2	3	3
Avg.	3	3	3	3	2	3	2	3

1-low, 2-medium, 3-high

CA24P23	**Full Stack Development Laboratory**	**Category**	L	T	P	C
		FC	0	0	4	2

PREREQUISITE

To get started with full-stack development, it's important to have a foundational understanding of both front-end and back-end development, along with some key tools and concepts. Knowledge of version control using Git, including basic commands like cloning, branching, committing, and merging. Understanding how to deploy applications to platforms like Heroku, AWS, or DigitalOcean**.**

Course Objectives :

1. Build Complete Applications.
2. Develop applications that are fully functional from the client side (front-end) to the server side (back-end) and database management.
3. Create intuitive and responsive user interfaces that enhance the user experience.
4. Develop APIs that allow communication between different parts of the application or with third-party services.
5. Encourage creativity and innovation in developing new features or improving existing ones.

LIST OF EXPERIMENTS

1.Working with git commands

2. Installation of Typescript

3. Programming with different data structures and functions using Typescript

4. Programming with classes and inheritance

5. Organization of the code with namespace

6. Packaging the code with added modules

7. Development of a web application using React.js

8. Development of a web application using Node.js

9. Development of a full stack web application

10. Deployment of web application using Docker

Total : 30 Periods

COURSE OUTCOMES: At the end of the course, the students will be able to:

COs	Course Outcome	Cognitive Level (K1 to K4)
CO1	Understand the programming skills in different parts of the application, including APIs, databases, and user interfaces.	*K1*
CO2	Write clean, maintainable, and modular code that can be easily updated or scaled as the application grows.	*K2*
CO3	Demonstrate robust server-side logic to handle client requests, process data, and manage application state.	*K4*
CO4	Develop and execute programs latest industry trends, tools, and best practices to improve the development process.	*K4*
CO5	Implement the problem-solving skills to debug and troubleshoot issues across the full stack.	*K3*

REFERENCES :

1. Mina Andrawos, Hands-On Full Stack Development with Go: Build full stack web applications with Go, React, Gin, and Gopher JS, Packt Publishing, First Edition, 2022.

2. Chris Northwood, The Full Stack Developer: Your Essential Guide to the Everyday Skills Expected of a Modern Full Stack Web Developer, Apress, Second Edition,2018.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	3	-	-	2
CO2	3	2	3	3	2	-	-	3
CO3	2	3	2	3	2	-	-	2
CO4	3	2	3	3	2	-	-	3
CO5	3	3	3	2	2	-	-	3
Avg.	3	3	3	3	2	-	-	3

1-low, 2-medium, 3-high

MA24E01	Probability and Statistics (Elective I)	Category	L	T	P	C
		Elective	3	0	0	3

PREREQUISITE:
The students should have basic knowledge in data collection, data analysis, data interpretation and research design.

OBJECTIVES:
The course will enable learners to:

- Understand the concepts of Data analysis.
- Apply the theory of probability and random variables to real world problems.
- Analyze the sampling distribution.
- Explain the inferences by using testing of hypothesis.
- Familiarize skills in Design of experiments, correlation and Regression analysis.

UNIT - I	EXPLORATORY DATA ANALYSIS	(9)

Definition of Statistics – applications - data types and measurements, graphical representation of data using histogram, line diagram, bar diagram, measures of central tendency and dispersion; coefficient of skewness and kurtosis.

UNIT - II	PROBABILITY AND RANDOM VARIABLES	(9)

Random experiment, sample space and events. Definitions of probability, addition and multiplication rules of probability, conditional probability. Random variables: pmf and pdf of random variables; Mathematical expectation: mean, variance, covariance, mgf and cgf of a random variable.

UNIT - III	SAMPLING DISTRIBUTIONS	(9)

Concepts of population, sample, parameter, statistic, and sampling distribution. Probability distributions: Binomial, Poisson and Normal distributions with their important characteristics.

UNIT - IV	TESTING OF HYPOTHESIS	(9)

Statistical hypotheses-Simple and composite, Statistical tests, Critical region, Type I and Type II errors, Testing of hypothesis – null and alternative hypothesis, level of significance. Test of significance using z, t, F and Chi-Square distributions.

UNIT - V	ADVANCED STATISTICAL METHODS	(9)

Analysis of one-way, two-way classifications and Latin Square Design. Correlation and regression analysis.

TOTAL: 45 PERIODS

COURSE OUTCOMES:
At the end of the course, the students will be able to:

COs	Course Outcome	Cognitive Level
CO1	Illustrate the data using graphical representation, central tendency.	Understand
CO2	Apply the concepts of random variable in mathematical expectation.	Apply
CO3	Learn to apply discrete and continuous distribution for various problems.	Analyze
CO4	Understand the concept of hypothesis testing.	Understand
CO5	Develop the skills in Design of Experiments, correlation and Regression analysis.	Analyze

TEXT BOOKS:

1. Gupta S.C & Kapoor V.K, “Fundamentals of Mathematical statistics”, Sultan Chand & sons, New Delhi, 2nd Edition, 2015.
2. Douglas C Montgomery, George C Runger, “Applied Statistics and Probability for Engineers”, Wiley student edition, Chennai, 2014.

REFERENCES:

1. Freund J.E, “Mathematical statistics”, Prentice Hall, New Delhi, 3rd Edition, 2016.

2. Levine, David M, Berenson, L Mark, Stephen, David, “Statistics for Managers Using Microsoft Excel”, PHI, New Delhi, 2nd Edition, 2016.

3. Murray Spiegel, John Schiller, and R. Alu Srinivasan, “Schaum's Outline of Probability and Statistics”, 3rd Edition, 2016.

4. Sheldon M. Ross, “Introduction to Probability and Statistics for Engineers and Scientists”, 4th Edition, 2016.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	2	2	-	-	2
CO2	3	3	3	2	2	-	-	2
CO3	3	3	3	2	2	-	-	2
CO4	3	3	3	2	2	-	-	2
CO5	3	3	3	2	2	-	-	2
Avg.	3	3	3	2	2	-	-	2

1-low, 2-medium, 3-high

CA24E01	TCP/IP (Elective I)	Category	L	T	P	C
		Elective	3	0	0	3

PREREQUISITE:
Networking fundamentals familiarity with basic networking terms like IP addresses, ports, protocols, and packet switching. Knowledge of Computer systems and computers communicate with each other and the basics of computer networking. Understanding the Data communication of how data is transmitted and received over a network.

OBJECTIVES:
The course will enable learners to:

1. Explain the basic concepts, protocols, and architecture of the TCP/IP suite.
2. Identify how data is transmitted over a network using TCP/IP protocols.
3. Set up and resolve common issues with TCP/IP configurations on various devices.
4. Recognize and explain the roles of key protocols such as IP, TCP, UDP, ICMP, DHCP, and DNS.
5. Apply knowledge of IP addressing, subnet masks, and CIDR notation to design and troubleshoot IP networks.

UNIT - I **INTRODUCTION** **(9)**

History –Standards–Internet–-OSImodel–Protocolsuite–Addressing–TransmissionMedia–LocalAreaand Wide Area Networks–Switching–Connecting devices–IP Addressing.

UNIT - II **INTERNET PROTOCOL** **(9)**

Sub netting–Supernetting–IPPackets–DeliveryandForwardingofIPPackets–Datagram–Fragmentation–Options-Checksum–ARP–RARP-InternetControlMessageProtocol–InternetGroupManagementProtocol.

UNIT - III **TCP&UDP** **(9)**

TCPServices–Features-Segmentation-TCPConnection-StateTransitionDiagram-WindowsinTCP-Flowcontrol – Error Control–CongestionControl–Timers-Package-UserDatagramprotocol– Services-Applications-Package.

UNIT - IV **APPLICATION LAYER AND CLIENT SERVER MODEL** **(9)**

Concurrency–BOOTP–DHCP–DomainNameSystem–NameSpace–Distribution–Resolution–Messages– Telnet–Rlogin–NetworkVirtualTerminal–CharacterSet–ControllingtheServer–RemoteLogin.

UNIT - V **APPLICATION PROTOCOLS** **(9)**

FileTransferProtocol–Connections –Communication–SimpleMailTransferProtocol–SimpleNetworkManagement Protocol–HyperTextTransferProtocol–Transaction–RequestandResponsemessages..

Total (L= 45, T = 0) = 45 Periods

Course Outcomes : At end of the course, the student will be able to

COs	**Course Outcome**	**Cognitive Level (K1 to K6)**
CO1	Recognize the network components, categories, topology and IP address and compare the ISO/OSI model with TCP/IP protocol suite.	**K2**
CO2	Discuss about the functionality of various internet protocols and gain the knowledge of the different routing protocols and algorithms.	**K6**
CO3	Appraise User datagram and transmission control protocols.	**K5**
CO4	Explain the knowledge of congestion control and QOS techniques and purpose of DNS and client - server model.	**K2**
CO5	Compare HTTP, HTTPs and FTP in world wide web.	**K5**

REFERENCES:

1. BehrouzA.Forouzan,TCP/IPProtocolSuite,TataMcGrawHill Edition, New Delhi, Third Edition, 2015.
2. Richard Stevens W. and Gabrani G., TCP/IP Illustrated Volume I, Pearson Education, New Delhi, 2019.
3. Douglas E. Comer, David L. Stevens, Internetworking with TCP/IP – Volume I,II,III, PHI Pvt. Ltd., Second Edition, 2015.
4. Tim Parker, Mark A., Sportack , TCP/IP Unleashed, Techmedia, New Delhi, Second Edition, 2016.
5. Douglas E. Comex, Internetworking with TCP/IP, Principles, protocols and architecture, PHI, New Delhi, Fifth Edition, 2016.
6. Behrouz A. Forouzan, TCP/IP Protocol Suite, Tata McGraw Hill, New Delhi, Third Edition, 2016.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	2	3	3	2	2	2	2
CO2	2	2	3	3	2	2	2	2
CO3	2	2	3	3	2	2	3	3
CO4	2	2	3	3	2	2	3	3
CO5	2	2	3	3	2	2	2	3
Avg.	2	2	3	3	2	2	2	3

1-low, 2-medium, 3-high

CA24E02	Unix And Network Programming (Elective I)	Category	L	T	P	C
		Elective	3	0	0	3

PREREQUISITE:
Proficiency in C is crucial for Unix and network programming. Many Unix systems and network applications are developed using C due to its close interaction with system-level operations. Familiarity with programming concepts such as variables, control structures (loops, conditionals), functions, and data structures. Basic knowledge of operating system concepts, such as processes, threads, memory management, and file systems, is important for understanding how Unix systems manage resources.

Course Objectives:

1. Understanding Unix Operating System Concepts : knowledge of the Unix system architecture, including processes, threads, and memory management.
2. Mastering Unix Programming : Learn to use Unix system calls for file manipulation, process control, and inter-process communication (IPC).
3. Network Programming Fundamentals : Learn the fundamentals of socket programming, including creating, binding, listening, accepting, and connecting sockets.
4.Developing Network Applications : Learn methods for serializing and deserializing data for transmission over a network.
5. Advanced Unix Programming Techniques : Explore various IPC mechanisms like pipes, message queues, semaphores, and shared

UNIT - I **INTRODUCTION&FILESYSTEM** **(9)**

OverviewofUNIXOS-FileI/O–FileDescriptors–FileSharing-FilesandDirectories–FileTypes-FileAccess Permissions–FileSystems–SymbolicLinks-StandardI/Olibrary–StreamsandFileObjects–Buffering-System Data Files and Information- Password File–Group File–Login Accounting–System Identification.

UNIT - II **PROCESSES** **(9)**

Environment of a UNIX Process – Process Termination – Command Line Arguments – Process Control – Process Identifiers - Process Relationships Terminal Logins – Signals –Threads.

UNIT - III **INTERPROCESSCOMMUNICATION** **(9)**

Introduction-MessagePassing(SVR4)-Pipes–FIFO–MessageQueues–Synchronization (SVR4)–Mutexes–Condition Variables–Read– WriteLocks–FileLocking–RecordLocking–Semaphores–SharedMemory(SVR4).

UNIT - IV **SOCKETS** **(9)**

Introduction–TransportLayer–SocketIntroduction-TCPSockets–UDPSockets-RawSockets–SocketOptions- I/O Multiplexing-Name and Address Conversions.

UNIT - V **APPLICATIONS** **(9)**

DebuggingTechniques-TCPEchoClientServer-UDPEchoClientServer-Ping-TraceRoute-ClientServer Applications Like File Transfer and Chat

Total (L= 45, T = 0) = 45 Periods

Course Outcomes : At end of the course, the student will be able to

Cos	***Course Outcomes***	**Cognitive Level (K1 to K6)**
CO1	Compare the system calls and library functions, different types of files and access permissions.	*K1*
CO2	Creation of parent and child process and gain the knowledge of the signals and threads.	*K4*

CO3	Identify the purpose of inter process communication system and locking procedure.	*K4*
CO4	Recognizing the different multiplexing techniques	*K1*
CO5	Appraise the TCP, UDP sockets and raw sockets.	*K3*

REFERENCES:

1. W.Richard Stevens, Advanced programming in the UNIX environment, Addison Wesley, New Delhi, 2015.
2. W.Stevens, Bill Fenner, Andrew Rudoff, Unix Network Programming, Volume 1,The Sockets Networking API, Pearson education, New Delhi, Third Edition, 2013.
3. W.Stevens, Bill F,A R, Unix Network Programming ,V1, The Sockets Networking API , PE,New Delhi, Third Edition, 2017.
4. Meeta G, Tilak S and Rajiv S The C Odyssey Unix –The open Boundless C , BPB Publications,New Delhi, First Edition, 2015.
5. W. Richard Stevens, Advanced Programming in The UNIX Environment, Addison Wesley, New Delhi 2015.
6. S. J. Leffler, M. K. Mckusick, M. J.Karels and J. S. Quarterman.,The Design and Implementation of the 4.3 BSD Unix Operating System, Addison Wesley,New Delhi, 2015.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	3	3	2	2
CO2	3	2	3	3	2	3	1	3
CO3	2	3	2	3	2	3	2	2
CO4	3	2	3	3	2	3	2	3
CO5	3	3	3	2	2	2	3	3
Avg.	3	3	3	3	2	3	2	3

1-low, 2-medium, 3-high

CA24E03	Web Programming Essential (Elective I)	Category	L	T	P	C
		Elective	3	0	0	3

PREREQUISITE:
Familiarity with using a computer, keyboard, and mouse. Understanding of basic software installation and management. Basic understanding of programming concepts (variables, data types, loops, conditional statements, functions). Familiarity with a programming language (e.g., Python, JavaScript, HTML/CSS).

Course Objectives:

1.Understand the basics of web development and the Internet

2.Understand JavaScript basics, data types, and control structures

3.Learn HTML5 structure, elements, and attributes

4.Develop a complete web application using client-side and server-side technologies

5.Explore new technologies and innovations in web programming

UNIT - I	WEBSITE BASICS	(9)

Internet Overview – Fundamental computer network concepts – Web Protocols – URL – Domain Name- Web Browsers and Web Servers- Working principle of a Website –Creating a Website – Client-side and server-side scripting

UNIT - II	WEB DESIGNING	(9)

HTML – Form Elements – Input types and Media elements – CSS3 – Selectors, Box Model, Backgrounds and Borders, Text Effects, Animations, Multiple Column Layout, User Interface.

UNIT - III	CLIENT-SIDE PROCESSING AND SCRIPTING	(9)

JavaScript Introduction – Variables and Data Types-Statements – Operators – Literals-Functions Objects-Arrays-Built-in Objects- Regular Expression, Exceptions, Event handling, Validation – JavaScript Debuggers.

UNIT - IV	SERVERSIDE PROCESSING AND SCRIPTING – PHP	(9)

PHP – Working principle of PHP – PHP Variables – Constants – Operators – Flow Control and Looping – Arrays – Strings – Functions – File Handling – File Uploading – Email Basics – Email with attachments – PHP and HTML – Simple PHP scripts – Databases with PHP.

UNIT - V	SERVLETS AND DATABASE CONNECTIVITY	(9)

Servlets: Java Servlet Architecture – Servlet Life cycle- Form GET and POST actions -Sessions – Cookies–Database connectivity–JDBC Creation of simple interactive applications – Simple database applications

Total (L: 45 T: 0) = 45 Periods

Course Outcomes : At end of the course, the student will be able to

COs	**Course Outcome**	**Cognitive Level (K1 to K6)**
CO1	Understand the basic concepts of website	**K2**
CO2	Describe the role of Web Designing.	**K2**
CO3	Explain the concepts of client-side processing and scripting.	**K2**
CO4	Identify and explain the key features of server side processing and scripting- php.	**K4**
CO5	Understand the basic concepts of servlets and database connectivity.	**K2**

REFERENCES:

1.	Robin Nixon, “Learning PHP, MySQL, JavaScript, CSS & HTML5” Third Edition, O’Reilly publishers, 2022.
2.	Paul Deitel, Harvey Deitel, Abbey Deitel, “Internet & World Wide Web – How to Program”, 5th edition, Pearson Education, 2019.
3.	Jeffrey C. Jackson, “Web Technologies–A Computer Science Perspective”, Pearson Education, 2016.
4.	James F. Kurose, “Computer Networking: A Top-Down Approach”, Sixth Edition, Pearson Education, 2012.
5.	Steven Holzemer , “PHP – The Complete Reference”, 1st Edition, Mc-Graw Hill, 2017.
6.	Fritz Schneider, Thomas Powell, “JavaScript – The Complete Reference”, 3rd Edition, McGraw Hill Publishers, 2017.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	2	3	2	3
CO2	3	2	3	3	2	3	2	2
CO3	3	2	3	3	2	3	2	2
CO4	3	2	3	3	2	3	2	2
CO5	3	2	3	3	2	3	2	3
Avg.	3	2	3	3	2	3	2	2

1-low, 2-medium, 3-high

CA24E04	Middleware Technology (Elective I)	Category	L	T	P	C
		Elective	3	0	0	3

Prerequisite:
Programming Fundamentals: Proficiency in at least one programming language (e.g., Java, C#, Python) to understand middleware components and their integration., Knowledge of operating system concepts, including processes, threads, and networking, is essential. Familiarity with networking concepts such as protocols, sockets, and communication methods is crucial for understanding how middleware facilitates communication between distributed systems.

Course Objectives:

1. Understand the fundamentals: Manages communication between distributed systems through message queues (e.g., RabbitMQ, Kafka). 2. Transaction Management: Understand concepts like ACID properties (Atomicity, Consistency, Isolation, Durability), transaction isolation levels, and mechanisms for handling concurrent transactions.
3. Object Middleware: Supports interaction between objects across different systems (e.g., CORBA).
4. Web Middleware: Provides services for web applications, including web servers and application servers (e.g., Apache Tomcat, Microsoft IIS).
5. Transaction Management: Middleware often manages distributed transactions to ensure data consistency across multiple systems.

UNIT - I **INTRODUCTIONCLIENT/ SERVER&MIDDLEWARETECHNOLOGY** **[09]**

Client/Server-ServerTypes-Middleware–Client,ServerandOperatingSystem-IntroductiontoDistributedObject Technology-Middleware–Client/ServerBuildingBlocks–Peer-toPeerCommunications–RPC-Messaging-JavaRMI-Overview of CORBA and DCOM.

UNIT - II **EJBARCHITECTURE** **[09]**

EJB – EJB Architecture – Overview of EJB Software Architecture – View of EJB –Conversation – Building and Deploying EJBs – Roles in EJB.

UNIT - III **EJBAPPLICATIONS** **[09]**

Types of Enterprise beans –Lifecycle of Beans-Steps in Developing an application using EJB Framework, EJB Deployment-EJB Session Beans– EJB Entity Beans–EJB Clients– Building an Application with EJB.

UNIT - IV **CORBA** **[09]**

Introduction and Concepts-CORBA Components-Architectural Features-Method Invocations-Static and Dynamic CORBA-StructureofCORBAIDL-SelfDescribingDatatypes-BuildinganapplicationusingCORBA-Advanced CORBA- CORBAObjectService-ObjectLocationService-MessageService-CORBAComponentModel.

UNIT - V **COM** **[09]**

Evolution of DCOM, COM Client and Server, COMIDL, COM Interface-COM threading Model, Marshalling, Comparison of RMI, CORBA and DCOM. Programming Examples of RMI, CORBA and DCOM.

Total 45 Periods

COURSE OUTCOMES*: At the edd of the course, the student will be able to:*

Cos	Course Outcomes	Cognitive Level (K1 to K6)
CO1:	Introduce the concept, techniques and applications of middleware technology.	*K2*
CO2:	Understand middleware components like COM, CORBA and EJB.	*K2*
CO3:	Explain about EJP Applications.	*K4*
CO4:	Illustrate the overview of CORBA concepts	*K4*
CO5:	Illustrate the overview of COM concepts	*K3*

Reference Books:

1	Herbert Schildt, The Complete Reference JAVA, Tata McGraw Hill, New Delhi, Tenth Edition, 2023.
2	Gavin King, Java Persistence with Hibernate, Manning Publications, New Delhi, Second Edition,2016.
3	Herbert Schildt, The Complete Reference, Tata McGraw Hill, New Delhi, Eighth Edition, 2016.
4	Kogent, Java 6 Programming Black Book, Kogent Learning Solutions, New Delhi, Second Edition, 2015,
5	Steven Holzner, Java 2 (JDK 5) Programming, PHI, New Delhi, Second Edition, 2017.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	3	3	2	2
CO2	3	2	3	3	2	3	1	3
CO3	2	3	2	3	2	3	2	2
CO4	3	2	3	3	2	3	2	3
CO5	3	3	3	2	2	2	3	3
Avg.	3	3	3	3	2	3	2	3

1-low, 2-medium, 3-high

CA24E05	Devops (Elective I)	Category	L	T	P	C
		Elective	3	0	0	3

Prerequisite:
To effectively work in a DevOps environment, having a strong foundation in several key areas is crucial. Understanding how to use Git for version control, including branching, merging, and pull requests. Knowledge of at least one language such as Python, Ruby, or JavaScript to write automation scripts or tools. Understanding how to create and manage Docker containers. Familiarity with services like compute instances, databases, and storage solutions provided by these platforms.

Course Objectives:

1. Understand the basic knowledge of micro services.
2. Understand the different types of micro services architecture.
3. Learn about basic concepts of Devops.
4. Learn about basic concepts of AWS, Azure, or Google Cloud Platform (GCP) for deploying and managing a applications and infrastructure in the cloud.
5. Implementing and tracking DevOps goals can transform how organization develops and delivers software.

UNIT - I	INTRODUCTION TO MICROSERVICES	(9)

Definition of Micro services – Characteristics - Micro services and Containers – Interacting with Other Services – Monitoring and Securing the Services – Containerized Services – Deploying on Cloud

UNIT - II	MICROSERVICES ARCHITECTURE	(9)

Monolithic architecture- Micro service architectural style- Benefits - Drawbacks of Micro service architectural style - decomposing monolithic applications into Micro services.

UNIT - III	BASICS OF DEVOPS	(9)

History of DevOps- DevOps and software development life cycle- water fall model – agile model – DevOps life cycle – DevOps tools: distributed version control tool –Git- automation testing tools – Selenium - reports generation – TestNG - User Acceptance Testing – Jenkins.

UNIT - IV	MICROSERVICES IN DEVOPS ENVIRONMENT	(9)

Evolution of Micro services and DevOps – Benefits of combining DevOps and Micro services working of DevOps and Microservices in Cloud environment - DevOps Pipeline representation for a NodeJS based Micro services.

UNIT - V	VELOCITY AND CONTINUOUS DELIVERY	(9)

Velocity - Delivery Pipeline- test stack - Small/Unit Test – medium /integration testing – system testing- Job of Development and DevOps - Job of Test and DevOps – Job of Op and Devops Infrastructure and the job of Ops.

Total 45 Periods

COURSE OUTCOMES*: At the edd of the course, the student will be able to:*

Cos	Course Outcomes	Cognitive Level (K1 to K6)
CO1:	Recognize the introduction about the Micro services	K2
CO2:	Discuss the concept of Micro services Architecture	K2
CO3:	Discuss the concept of Devops	K4
CO4:	Describe DevOps and the common tools used in DevOps	K4

CO5:	Apply Micro services in DevOps	K3

Reference Books:

1	James A Scott, A Practical Guide to Microservices and Containers, Map R Data Technologies e–book.
2	Joyner Joseph, Devops for Beginners, Mihails Konoplovs publisher, First Edition, 2015.
3	Gene Kim, Kevin Behr, George Spafford, The Phoenix Project, A Novel about IT, DevOps, IT Revolution Press, 5th Edition, 2018..
4	Michael Hüttermann, DevOps for Developers, APress, e-book, First Edition, 2012.
5	Jennifer Davis and Katherine Daniels, Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale, O'Reilly Media Inc, First Edition, 2016.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	3	3	2	2
CO2	3	2	3	3	2	3	1	3
CO3	2	3	2	3	2	3	2	3
CO4	3	2	3	3	3	3	2	3
CO5	3	3	3	2	2	2	3	3
Avg.	3	2	3	3	2	3	2	3

1-low, 2-medium, 3-high

BA24E51	Health Care Information Systems (Elective I)	Category	L	T	P	C
		Theory	3	0	0	3

PREREQUISITE: The students should have basic knowledge on the concepts of healthcare, information system, information technology and related basic laws.

OBJECTIVES

- To understand the basic concepts of healthcare information and law.
- To identify the current scenario of healthcare information systems.
- To recognize the various management information technology challenges.
- To facilitate the different standardsin managing the information technology.
- To explore the various initiatives involved in healthcare information system.

UNIT - I	INTRODUCTION	[09]

Introduction to Healthcare Information – Health Care Data Quality - Healthcare Information Regulations, Laws And Standards

UNIT - II	HEALTHCARE INFORMATION SYSTEMS	[09]

History and Evolution of Healthcare Information Systems - Current and Emerging use of Clinical Information Systems - System Acquisition - System Implementation and Support.

UNIT - III	INFORMATION TECHNOLOGY	[09]

Information Architecture and Technologies that Support Health Care Information Systems - Health Care Information Systems Standards - Security of Healthcare Information Systems.

UNIT - IV	MANAGEMENT OF IT CHALLENGES	[09]

Organizing Information Technology Services - IT Alignment and Strategic Planning - IT Governance and Management.

UNIT - V	IT INITIATIVES	[09]

Management’s Role in Major IT Initiatives - Assessing and Archiving Value in Healthcare Information Systems.

TOTAL: 45 PERIODS

COURSE OUTCOMES:
At the end of the course, the students will be able to:

Cos	Course Outcomes	Cognitive Level (K1 to K6)
CO1	Assimilate the concept of major types of healthcare information	Understand
CO2	Acquire knowledge on healthcare information systems.	Remembering
CO3	Identify the concept of information technology and its uses.	Remembering
CO4	assess the information technology adaption in management and IT Governance.	Understand
CO5	Familiarize with the concept of IT initiatives**.**	Analyze

REFERENCES:

1.Aswathappa.K, Organizational Behavior, Himalaya Publishing House, Chennai, Tenth Edition, 2015.

2.Stephen P. Robins, Organizational Behavior, PHI Learning / Pearson Education, New Delhi, Eleventh edition, 2016.

3.Stephen P Robbins, Organizational Behavior, PHI, New York, Thirteen Edition, 2014

4.MohiniSukhapure&UdayN.Limaye Organizational Behavior, Himalaya Publishing Private Limited, Pune, 2015

5.P.Subba Rao, Organizational Behavior, Himalaya Publishing (P) Ltd, Pune, Fifth Edition, 2016

6.Schermerhorn, Hunt and Osborn, Organizational behavior, John Wiley, New Delhi, Ninth Edition, 2015

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	2	2	2	3	2	2
CO2	3	2	2	2	2	3	2	2
CO3	3	2	2	2	2	3	2	2
CO4	3	2	2	2	2	3	2	2
CO5	3	2	2	2	2	3	2	2
Avg.	3	2	2	2	2	3	2	2

1-low, 2-medium, 3-high

MA24E02	Resource Management Techniques (Elective II)	Category	L	T	P	C
		Theory	3	0	0	3

PREREQUISITE

For Effective learning and applying resource management technique students must have a foundational understanding of optimization technique like linear programming and integer programming, basic knowledge of network programming, Replacement model.

OBJECTIVES:

The course will enable learners to:

- To enable learners to determine the most effective way to allocate the best value of linear programming such as profit or loss based on decision variables.
- To make the learners how to analyze the most effective way to minimize the total transportation cost and to find the optimal way to assign a set of tasks.
- To facilitate learners about the network models to identify shortest path, Network design and Project Scheduling.
- To provide the learners how to determine optimal quantity of inventory to hold, balancing between excess and shortage improve optimal efficiency and reduce wastage.
- To facilitate learners to determine cost reduction related to maintenance, downtime replacement of assets, scheduling tasks, production runs in optimal order.

UNIT - I | LINEAR PROGRAMMING PROBLEMS | (9)

Introduction–Scope and role of OR–Phases of OR–Limitations of OR–Linear Programming Problem–Formulation of Linear Programming Problem–Optimum Solution by Graphical Method–Simplex Method(Using Slack Variables).

UNIT - II | TRANSPORTATION AND ASSIGNMENT PROBLEMS | (9)

Transportation Models(Minimizing and Maximizing Cases)-Balanced and Unbalanced Cases–Initial Basic Feasible Solution by North West Corner Rule, Least Cost and Vogel’s Approximation Methods. Check for Optimality by Modified Method Assignment Models.

UNIT - III | NETWORK MODELS | (9)

Network – Fulkerson’s Rule – Construction of a Network – Critical Path Method (CPM) – Optimistic, Pessimistic and Most Likely Time Estimates – Project Scheduling by PERT Analysis.

UNIT - IV | INVENTORY MODEL | (9)

Types of Inventory – Deterministic Inventory Models – EOQ and EBQ Models with and without Shortages – Quantity Discount Models - Price Breaks.

UNIT - V | REPLACEMENT MODELS AND SEQUENCING | (9)

Replacement of items that Deteriorate with Time – Value of Money Changing with Time–Not Changing with Time– Optimum Replacement Policy – Individual and Group Replacement. Sequencing Problem–Assumptions – Processing of ‘n’ Jobs with two Machines ’n’ Jobs with ‘m’ Machines.

TOTAL: 45 PERIODS

COURSE OUTCOMES: At the end of the course, the students will be able to:

Cos	Course Outcomes	Cognitive Level (K1 to K6)
CO1	Apply the concepts of linear programming approach during the uncertain situations.	Apply
CO2	Analyze the transportation method and Assignment method to minimize costs	Analyze
CO3	Construct and Apply the concepts of network model to identify shortest path.	Apply
CO4	Evaluate the inventory model using EOQ and EBQ with and without shortage.	Evaluate
CO5	Analyze the replacement models to get optimum production.	Analyze

TEXT BOOKS:

1. TahaH.A, “Operation Research”, Pearson Education, Noida , 9th Edition, 2013

2.Vohra N D, “Quantitative Techniques in Management”, Tata Mc Graw Hill, New Delhi, 6th Edition, 2021.

REFERENCES:

1.P.K.Gupta and Man Mohan, “Problems in Operations Research”, S.Chand and Co, New Delhi, 12th Edition, 2014

2. Wayne. L. Winston, “Operations research applications and algorithms”, Thomson learning, United States, 4th Edition,2016.

3. Kalavathy S, “Operations Research”, Vikas Publishing House, Ahmedabad, 6th Edition, 2019.

4. Hiraand Gupta, “Problems in Operations Research”, S.Chand and Co, New Delhi, 2nd Edition, 2012.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	2	2	-	-	2
CO2	3	3	3	2	2	-	-	2
CO3	3	3	3	2	2	-	-	2
CO4	3	3	3	2	2	-	-	2
CO5	3	3	3	2	2	-	-	2
Avg.	3	3	3	2	2	-	-	2

1-low, 2-medium, 3-high

CA24E06	Software Project Management (Elective II)	Category	L	T	P	C
		Elective	3	0	0	3

PREREQUISITE

Understand the phases of software development, including planning, analysis, design, implementation, testing, and maintenance. Familiarity with project management concepts, such as scope, schedule, budget, resources, and risk management. Understanding of software design patterns, coding standards, and testing methodologies.

Course Objectives :

1. Define software project management, its importance, and its relationship to software development life cycles.
2. Understand leadership and team management concepts, including team formation, roles, and responsibilities.
3. Apply risk management techniques, including risk identification, analysis, prioritization, and mitigation.
4. Develop communication plans and apply effective communication techniques for managing stakeholder expectations.
5. Evaluate and improve software project management processes using metrics, benchmarks, and best practices.

UNIT - I INTRODUCTION TO SOFTWARE PROJECT MANAGEMENT (9)

Product Life Cycle: Introduction-Idea Generation -Prototype Development Phase-Alpha Phase - Beta Phase - Production Phase-Maintenance and Obsolescence Phase. Product Life Cycle Models: The Waterfall Model - The Prototyping Model - The Rapid Application Development (RAD) Model - Spiral Model and Its Variants. Process Models: The ISO-9001 Model - The Capability Maturity Model.

UNIT - II SOFTWARE METRICS & SOFTWARE CONFIGURATION MANAGEMENT (9)

Software Metrics: Introduction - The Metrics Roadmap - A Typical Metrics Strategy - What To Measure - Set Targets and Track Them - Understanding and Trying to Minimize Variability - Act on Data - People and Organizational Issues in Metrics Programs - Common Pitfalls to watch out for in Metrics Programs – Metrics Implementation Checklists and Tools. Software Configuration Management: Introduction - Definitions and Terminology

UNIT - III SOFTWARE QUALITY ASSURANCE & RISKMANAGEMENT (9)

Software Quality Assurance: Software Quality - Quality Important in Software - Quality Control and Quality Assurance -Cost and Benefits of Quality - Software Quality Analyst's Functions - Misconceptions about the SQA Role - Software Quality Assurance Tools- Organizational Structures - Profile of a Successful SQA - Measure of SQA Success - Pitfalls of SQA. Risk Management: Introduction

UNIT - IV PROJECT MANAGEMENT PROCESS AND ACTIVITIES (9)

Project Life Cycle: In-Stream Activities-Project Initiation: Activities During Project Initiation - Outputs, Quality Record and Project Initiation Phase – Interface to the Process Database. Project Planning and Tracking: Components of Project Planning and Tracking - Project Closure: Issues and Metrics for Project Closure.

UNIT - V ENGINEERING ACITIVITIES IN PROJECTS (9)

Estimation: Phases of Estimation - Estimation Methodology - Size Estimation Effort and Schedule Estimates. Project Management In Testing Phase - What Is Testing? – What are the Activities That Make Up Testing - Test Scheduling & Types of Tests - Project Management In the Maintenance Phase - Activities During the Maintenance Phase - Management Issues in the Maintenance Phase.

Total (L: 45 T: 0) = 45 Periods

Course Outcomes : At the end of the course, the student will be able to:

COs	Course Outcome	Cognitive Level (K1 to K6)
CO1	Identify the basic idea about the software project management and it's planning activities.	**K3**
CO2	Understand the software metrics and SCM.	**K2**
CO3	Understand the role of software developers in getting exposure on software quality and risk management.	**K2**

CO4	Understand the project management process and its activities.	**K2**
CO5	Create a project reporting, defect analysis and prevention.	**K6**

REFERENCES:

1. GopalaswamyRamesh,ManagingGlobalSoftwareProjects,TataMcGrawHill, New Delhi, Third Edition,2018
2. Pankaj Jalote, Software Project Management in Practice, Pearson Education, New Delhi, Second Edition 2018
3. BobHughes,Mikecotterell,RajibMall,SoftwareProjectManagement,FifthEdition, TataMcGrawHill,2011.
4. WalkerRoyce,SoftwareProjectManagement,PearsonEducation,New Delhi, Second Edition 2017.
5. PankojJalote,SoftwareProjectManagementin Practice,PearsonEducation, Chennai,2018.
6. Jim Highsmith, Agile Project Management, Pearson education, New Delhi, 2018.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	2	2	3	3	2
CO2	3	3	3	2	2	3	3	2
CO3	3	3	3	2	2	3	3	2
CO4	3	3	3	2	2	3	3	2
CO5	3	3	3	2	2	3	3	2
Avg.	3	3	3	2	2	3	3	2

1-low, 2-medium, 3-high

CA24E07	Advanced Operating Systems (Elective II)	Category	L	T	P	C
		Elective	3	0	0	3

PREREQUISITE

Understanding core concepts such as processes, threads, scheduling, memory management, and file systems. Familiarity with system calls and how operating systems interact with hardware and applications. Knowledge of computer hardware, including CPUs, memory hierarchies, and I/O systems. Understanding of virtual memory management, paging, and segmentation. Knowledge of process creation, management, and synchronization mechanisms. Basics of resource allocation, deadlock, and resource management strategies.

Course Objectives:

1. Advanced System Architectures : understanding of various operating system architectures, including monolithic kernels, micro kernels, and hybrid systems.

2. Memory Management : Learn about advanced memory management techniques, including virtual memory, paging, segmentation, and demand paging.

3. Process and Thread Management: Explore sophisticated scheduling algorithms and techniques, including multi-level feedback queues, real-time scheduling, and load balancing.

4. File Systems : Study advanced file system designs, including journaling, log-structured file systems, and distributed file systems.

5. Distributed Systems : Learn about the principles of distributed operating systems, including process distribution, inter-process communication, and consistency models.

UNIT - I **DISTRIBUTEDOPERATINGSYSTEMS** **(9)**

Architectures of Distributed Systems-System Architecture Types-Issues in Distributed Operating Systems-Communication Networks-Communication Primitives. Distributed Dead Lock Detection-Introduction-Deadlock HandlingStrategiesinDistributedSystems-IssuesinDeadlockDetectionandResolution –Control Organizations for Distributed Deadlock Detection.

UNIT - II **DISTRIBUTEDRESOURCEMANAGEMENT** **(9)**

DistributedFileSystems–MechanismsforBuildingDistributedFileSystems–DesignIssues–DistributedShared Memory–DesignIssues:DistributedScheduling–IssuesinLoadDistributing–ComponentsofaLoadDistributing Algorithm–LoadDistributingAlgorithms–SelectingaSuitableLoadSharingAlgorithm–Requirements forLoad Distributing.

UNIT - III **FAILURERECOVERYANDFAULTTOLERANCE** **(9)**

Recovery–BasicConcepts–ClassificationofFailures–BackwardandForwardErrorRecovery–Backward-Error Recovery:BasicApproaches–RecoveryinConcurrentSystems–FaultTolerance–Issues–AtomicActionsand Committing–CommitProtocols–NonblockingCommitProtocols–VotingProtocol

UNIT - IV **PROTECTIONANDSECURITY** **(9)**

ProtectionandSecurity-Preliminaries,TheAccessMatrixModelanditsimplementations.-Safety in Matrix Model- Advanced Models of Protection

UNIT - V **MULTIPROCESSOROPERATINGSYSTEMS** **(9)**

Multiprocessor Operating Systems-Basic Multiprocessor System Architectures –Inter Connection Networks forMultiprocessorSystems-Caching-HypercubeArchitecture-StructuresofMultiprocessorOperatingSystem-Operating System Design Issues.

Total (L: 45 T: 0) = 45 Periods

Course Outcomes : At the end of the course, the student will be able to:

Cos	Course Outcomes	Cognitive Level (K1 to K6)
CO1	Illustrate comprehensive knowledge of the architecture of distributed systems.	K2
CO2	Understand the deadlock and their solutions in distributed environments,	K3
CO3	Analyze the knowledge of failure recovery and fault tolerance, to know the security issues and protection mechanisms for distributed environments,	K6

CO4	Understand the multiprocessor operating systems	K2
CO5	Understand the main concepts of advanced operating systems	K3

REFERENCES:

1. MukeshSinghal, NiranjanG.Shivaratri, Advanced concepts in operating systems: Distributed, Database and multiprocessor operating systems, Tata McGraw Hill, New Delhi, Second Edition, 2017.
2. Abraham Silberschatz; Peter Baer Galvin; Greg Gagne, Operating System Concepts, John Wiley and Sons, New York, Seventh Edition, , 2014
3. AndrewS.Tanenbaum,Modernoperatingsystem,PHI, New York,2015
4. PradeepK.Sinha,DistributedoperatingsystemConceptsanddesign,PHI, New Delhi,2015.
5. AndrewS.Tanenbaum,Distributedoperatingsystem,Pearsoneducation, New York, 2015.
6. S. Tanenbaum and A. S. Woodhull, Operating Systems Design and Implementation, Prentice Hall, London, Third Edition, 2016.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	3	3	2	2
CO2	3	2	3	3	2	3	1	3
CO3	2	3	2	3	2	3	2	2
CO4	3	2	3	3	2	3	2	3
CO5	3	3	3	2	2	2	3	3
Avg.	3	3	3	3	2	3	2	3

1-low, 2-medium, 3-high

CA24E08	E-Learning Techniques (Elective II)	Category	L	T	P	C
		Elective	3	0	0	3

PREREQUISITE

Understanding of fundamental instructional design theories and models, such as Analysis, Design, Development, Implementation, Evaluation. Proficiency with using computers, operating systems, and common software applications. Knowledge of popular LMS platforms such as Moodle, Blackboard, Canvas, or Google Classroom, including their features and functionalities. Understanding of multimedia elements such as text, images, audio, and video, and how they are used in e-learning.

Course Objectives:

1. E-Learning Fundamentals : Understand the key concepts and principles of e-learning, including its history, types, and benefits.

2. Content Development and Organization : Develop instructional materials that are engaging, relevant, and pedagogically sound, using text, images, audio, and video.

3. E-Learning Tools and Platforms : Navigate and use LMS platforms to manage courses, track learner progress, and facilitate interactions.

4. Multimedia and Interactive Elements : Incorporate multimedia elements like videos, animations, and simulations to enhance learning.

5. User Experience (UX) and Usability : Design e-learning environments with a focus on usability, accessibility, and user experience.

UNIT - I	INTRODUCTION	(9)

What is E-Learning – E-Learning Evolution – Advantages and Disadvantages of E-Learning – Instructional Design Models for E-Learning – Applying User – Centered Design to E-Learning – Rapid E-Learning.

UNIT - II	KEEPING THE E-LEARNING STRATEGY FORWARD	(9)

Learning Strategy – Process for Developing the E-Learning Strategy – Doomed to Failure – Keeping Focused on the Strategy – Instructional Strategies for E-Learning.

UNIT - III	DELIVERING E-LEARNING &E-LEARNING EVALUATION	(9)

Delivering E-Learning – Instructional Game Characteristics – Educational Podcasting - Gaming at Work – Delivering E-Learning Synchronously – E-Learning Education – Four Levels of Evaluating Learning – Learning Analytics - Evaluation Models.

UNIT - IV	WEB STANDARDS	(9)

Resources for Guidance on Web Standards - Web Standards for Designers – Validators - W3C Keeping it Simple.

UNIT - V	E-LEARNING TOOLS	(9)

E-learning Tools – E-learning Authoring Tools – Wikis and E-Learning.

Total (L: 45 T: 0) = 45 Periods

Course Outcomes : At the end of the course, the student will be able to:

Cos	*Course Outcomes*	Cognitive Level (K1 to K6)
CO1	Understand the concept of e-learning.	*K2*
CO2	Understand the concept of various learning strategies.	*K3*
CO3	Express the concept of delivering e-learning.	*K6*
CO4	Understand the concept of web standards for designing, valuators.	*K2*
CO5	Understand the knowledge about the e-learning tools.	*K3*

REFERENCES:

1. Randy Garrison D ,E-Learning in the 21st century a framework for research and practice, Second edition, Taylor and Francis, 2015.
2. Robin Mason,E-Learning : the key concepts, Routledge, 2015.
3. Clark R.C and Mayer R.E, E-Learning and the science of instruction, Pfeiffer Wiley, 2015.

4. Mark J Rosenberg, E-Learning: strategies for delivering knowledge in the Digital Age, McGraw- Hill, New Delhi, 2016.
5. Kjell E. (Erik) Rudestam , Judith Schoenholtz - Read, Handbook of Online Learning, Sage Publications Inc., London, Second Edition, 2019.
6. John Gardner, Bryn Holems, E-Learning: Concepts and practice, SAGE Publications, New Delhi 2016.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	3	3	2	2
CO2	3	2	3	3	2	3	1	3
CO3	2	3	2	3	2	3	2	2
CO4	3	2	3	3	2	3	2	3
CO5	3	3	3	2	2	2	3	3
Avg.	3	3	3	3	2	3	2	3

1-low, 2-medium, 3-high

CA24E09	Soft Computing (Elective II)	Category	L	T	P	C
		Elective	3	0	0	3

Prerequisite:

Programming Fundamentals: Knowledge of fundamental algorithms and data structures is essential for implementing and understanding soft computing methods., Proficiency in a programming language (e.g., Python, MATLAB, Java) to implement and experiment with soft computing algorithms. Understanding of basic computer science principles, including computer architecture and software development. Familiarity with basic AI concepts, such as search algorithms, neural networks, and machine learning, which are often used in soft computing.

Course Objectives:

1. Understand the fundamentals: Inspired by biological neural networks, these systems learn from data to recognize patterns and make decisions. Key concepts include neurons, activation functions, and training algorithms like back propagation.
2. Genetic Algorithms: Optimization techniques inspired by natural selection, involving processes like mutation, crossover, and selection to evolve solutions to problems.
3. Evolutionary Computation: Includes genetic algorithms and other techniques that use evolutionary processes to solve optimization problems.
4. Applications: Soft computing methods are used in various fields such as control systems, data mining, pattern recognition, decision-making, and optimization problems.
5. Hybrid Systems: Often, soft computing techniques are combined with traditional methods (e.g., fuzzy-neural hybrids) to leverage the strengths of both approaches for more robust solutions.

UNIT - I INTRODUCTIONTO SOFTCOMPUTINGANDNEURAL NETWORKS [09]

EvolutionofComputing-SoftComputingConstituents–FromConventionalAItoComputationalIntelligence -Machine Learning Basics.

UNIT - II GENETIC ALGORITHMS [09]

Introduction to Genetic Algorithms (GA) – Applications of GA in Machine Learning - Machine Learning Approach to Knowledge Acquisition.

UNIT - III NEURALNETWORKS [09]

Machine Learning using Neural Network, Adaptive Networks –Feed Forward Networks Supervised Learning Neural Networks–Radial Basis Function Networks-Reinforcement Learning–Unsupervised Learning Neural Networks– Adaptive Resonance Architectures–Advances in Neural networks.

UNIT - IV FUZZYLOGIC [09]

FuzzySets–OperationsonFuzzySets–FuzzyRelations–MembershipFunctions –Fuzzy Rules and Fuzzy Reasoning – Fuzzy Inference Systems–Fuzzy Expert Systems–Fuzzy Decision Making.

UNIT - V NEURO-FUZZY MODELING [09]

Adaptive Neuro - Fuzzy Inference Systems–Coactive Neuro - Fuzzy Modeling–Classification and Regression Trees– Data Clustering Algorithms–Rule Base Structure Identification–Neuro – Fuzzy Control.

Total 45 Periods

COURSE OUTCOMES*: At the edd of the course, the student will be able to:*

Cos	***Course Outcomes***	**Cognitive Level (K1 to K6)**
CO1:	Understand the concept of soft computing.	K2
CO2:	Understand the concept of genetic algorithm.	K2
CO3:	Understand the concept of neural networks basic.	K4
CO4:	Understand the concept of fuzzy modeling.	K4
CO5:	Develop the knowledge about the neuro-fuzzy modeling.	K3

Reference Books:

1	Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, Neuro-Fuzzy and Soft Computing, PHI, New Delhi 2023.
2	James A., Freeman & David M,Skapura, Neural Networks Algorithms Applications & Prg. Tech., PE,Chennai 2016
3	MitchellMelanie,AnIntroductiontoGeneticAlgorithm,PrenticeHall,New Delhi2013.
4	Sivanandam, S. N., Sumathi, S., and Deepa, S.N., Introduction to Fuzzy Logic using MATLAB, Springer, 2017.
5	Sivanandam, S.N., Deepa, S.N., Introduction to Genetic Algorithms, Springer, 2017.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	3	3	2	2
CO2	3	2	3	3	2	3	1	3
CO3	2	3	2	3	2	3	2	2
CO4	3	2	3	3	2	3	2	3
CO5	3	3	3	2	2	2	3	3
Avg.	3	3	3	3	2	3	2	3

1-low, 2-medium, 3-high

CA24E10	Object Oriented Analysis And Design (Elective II)	Category	L	T	P	C
		Elective	3	0	0	3

Prerequisite:

Understanding at least one object-oriented programming language is crucial. Familiarity with concepts like classes, objects, inheritance, encapsulation, and polymorphism will be beneficial. Knowledge of basic software engineering principles, such as requirements gathering, design methodologies, and testing. A grasp of fundamental data structures and algorithm is important as they are often used in the design and implementation phases.

Course Objectives:

1. Understand Object-Oriented Concepts: Gain a thorough understanding of fundamental object-oriented principles such as classes, objects, inheritance, encapsulation, and polymorphism.

2. Apply OOAD Methodologies: Learn to apply the principles of object-oriented analysis and design in real-world scenarios.

3. Develop Use Cases: Understand how to identify and define use cases to capture functional requirements.

4. Design and Refactor Object-Oriented Systems: Apply design principles and patterns to create robust and maintainable object-oriented systems.

5. Evaluate and Test Object-Oriented Designs: Learn techniques for evaluating the quality and effectiveness of object-oriented designs.

UNIT - I UNIFIED PROCESS AND USE CASE DIAGRAMS (9)

Introduction to OOAD with OO Basics - Unified Process – UML diagrams – Use Case –Case study - the Next Gen POS system, Inception -Use case Modeling – Relating Use cases – include, extend and generalization – When to use Use-cases.

UNIT - II STATIC UML DIAGRAMS (9)

Class Diagram–– Elaboration – Domain Model – Finding conceptual classes and description classes – Associations – Attributes – Domain model refinement – Finding conceptual class Hierarchies – Aggregation and Composition - Relationship between sequence diagrams and use cases – When to use Class Diagrams.

UNIT - III DYNAMIC AND IMPLEMENTATION UML DIAGRAMS (9)

Dynamic Diagrams – UML interaction diagrams - System sequence diagram – Collaboration diagram – When to use Communication Diagrams - State machine diagram and Modeling –When to use State Diagrams - Activity diagram – When to use activity diagrams - Implementation Diagrams - UML package diagram - When to use package diagrams - Component and Deployment Diagrams – When to use Component and Deployment diagrams.

UNIT - IV DESIGN PATTERNS (9)

GRASP: Designing objects with responsibilities – Creator – Information expert – Low Coupling – High Cohesion – Controller - Design Patterns – creational – factory method – structural – Bridge – Adapter – behavioral – Strategy – observer – Applying GoF design patterns – Mapping design to code.

UNIT - V TESTING (9)

Object Oriented Methodologies – Software Quality Assurance – Impact of object orientation on Testing – Develop Test Cases and Test Plans.

Total 45 Periods

COURSE OUTCOMES*: At the edd of the course, the student will be able to:*

Cos	***Course Outcomes***	**Cognitive Level (K1 to K6)**
CO1:	*Summarize the Concepts of unified process and use case Diagrams.*	*K2*
CO2:	*Explain about the Static UML Diagrams.*	*K3*
CO3:	*Illustrate the Dynamic and implementation UML Diagrams..*	*K6*
CO4:	*Summarize the Design Patterns.*	*K2*
CO5:	*Explain the concepts of Testing in OOAD.*	*K3*

Reference Books:

1	Craig Larman, Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, Pearson Education – Third Edition, 2015
2	Ali Bahrami - Object Oriented Systems Development - McGraw Hill International Edition – Re-Print 2018.
3	Erich Gamma, and Richard Helm, Ralph Johnson, John Vlissides, ―Design patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, 2015
4	Martin Fowler,UML Distilled: A Brief Guide to the Standard Object Modeling Language‖, Addison Wesley, Third Edition, 2013
5	Deital& Deital,InternetandWorldWideWeb–Howtoprogram,Pearson,2011
6	Margaret Levine Young and Doug Muder, Internet: The Complete Reference McGraw Hill International Edition, First Edition, 2011

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	2	3	2	2
CO2	3	2	3	3	2	3	1	3
CO3	2	3	2	3	2	3	2	2
CO4	3	2	3	3	2	3	2	3
CO5	3	3	3	2	2	3	3	3
Avg.	3	3	3	3	2	3	2	3

1-low, 2-medium, 3-high

BA24E52	Organizational Behavior (Elective II)	Category	L	T	P	C
		Theory	3	0	0	3

PREREQUISITE: The students should have the basic knowledge on the concept of organizational behavior, workplace, personality, group behavior, leadership style, delegation of power and work life balance.

OBJECTIVES:

- To familiarize the essential of behavior influence in the workplace and the various models of behavior.
- To identify the individual behavior characteristics and its impact.
- To imbibe the group behavior and its dynamics.
- To assimilate importance of leadership styles, power and politics in organizations.
- To explore work life balance by managing stress and understanding different types of culture.

UNIT - I INTRODUCTION [09]

Organizational behavior: Definition – Meaning – Scope & Importance of OB – OB Model. Management and Managers: Functions – Skills – Roles – Types of Managers.

UNIT - II INDIVIDUAL BEHAVIOUR [09]

Personality: Theories – Types. Learning: Meaning and Definition – Theories of Learning. Attitudes: Nature – Components – Formation – Functions – Measurement. Perception: Factors Influencing Perception. Motivation: Importance – Theories (Maslow's Hierarchy Theory & Herzberg Theory) – Types – Effects on Work Behavior

UNIT - III GROUP BEHAVIOUR [09]

Nature – Types – Group Development – Group behavior – Structuring. Group Decision Making Techniques. Team Dynamics: Nature of Teams – Teams Vs Groups – Benefits From Teams – Types of Teams – Team Issues – Effective Team Work.

UNIT - IV LEADERSHIP AND POWER [09]

Leadership: Meaning – Importance – Leadership Styles – Theories. Power: Power Dynamics – Sources of Power – Effective Use of Power.

UNIT - V DYNAMICS OF ORGANIZATIONAL BEHAVIOUR [09]

Resistance of Change – Managing Change. Job Satisfaction: Determinants – Measurements. Stress: Prevention and Management of Stress – Balancing Work and Life.

TOTAL: 45 PERIODS

COURSE OUTCOMES:
At the end of the course, the students will be able to:

COs	Course Outcome	Cognitive Level (K1 to K4)
CO1	Explain The Concepts Of Organizational Behavior And The Various Models Of Behavior.	Remembering
CO2	Enhance Their Individual Behavior By Fine Tuning Personality, Emotions, Perceptions And Attitude In The Work Environment.	Understand
CO3	To Get Knowledge On The Concept Of Perception, Motivation Team Work, Leadership And Power In The Present Work	Analyze
CO4	Exercise Leadership Styles By Utilizing The Authority In An Ethical Manner In The Work Place.	Analyze

CO5	Illustrate The Concept Of Dynamic Of Organizational Behavior.	Understand

REFERENCES:

1. Aswathappa.K, Organizational Behavior, Himalaya Publishing House, Chennai, Tenth Edition, 2015.
2.Stephen P. Robins, Organizational Behavior, PHI Learning / Pearson Education, New Delhi, Eleventh edition, 2016.
3.MohiniSukhapure&UdayN.Limaye Organizational Behavior, Himalaya Publishing Private Limited, Pune, 2015
4. P.Subba Rao, Organizational Behavior, Himalaya Publishing (P) Ltd, Pune, Fifth Edition, 2016
5. Schermerhorn, Hunt and Osborn, Organizational behavior, John Wiley, New Delhi, Ninth Edition, 2015

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	2	2	2	3	2	2
CO2	3	2	2	2	2	3	2	2
CO3	3	2	2	2	2	3	2	2
CO4	3	2	2	2	2	3	2	2
CO5	3	2	2	2	2	3	2	2
Avg.	3	2	2	2	2	3	2	2

1-low, 2-medium, 3-high

K.S.R. COLLEGE OF ENGINEERING (Autonomous)					R 2024
SEMESTER - I					
BC5001	**Computer Fundamentals**	L	T	P	C
		2	0	0	0

Prerequisite: - Basic computer literacy: Students should have basic computer skills, including typing, using a mouse, and navigating a computer desktop.No prior programming experience required: Computer Fundamentals courses are often designed for beginners, so no prior programming experience is required.Basic understanding of technology: Students should have a basic understanding of technology concepts, including hardware, software, and networking.

Course Objectives:

1. Understand computer hardware components: Students will understand the basic computer hardware components, including CPU, memory, storage, and input/output devices.
2. Understand computer software concepts: Students will understand the basics of computer software, including operating systems, productivity software, and web browsers.
3. Understand networking fundamentals: Students will understand basic networking concepts, including LAN, WAN, and internet.
4. Understand cybersecurity basics: Students will understand basic cybersecurity concepts, including security threats and best practices.
5. Develop basic computer skills: Students will develop basic computer skills, including typing, using a mouse, and navigating a computer desktop.

UNIT - I INTRODUCTION [04]

Computer - Characteristics of Computers, Input, Output, Storage units, CPU, Computer System. Computer Organization Central Processing Unit - Processor Speed, Cache, Memory, RAM, ROM, Booting, Generation of

UNIT - II MEMORY-DEVICES [04]

Memory- Secondary Storage Devices: Floppy and Hard Disks, Optical Disks CD-ROM, DVD, Mass Storage Devices: USB thumb drive. Managing disk Partitions, File System Input. Devices - Keyboard, Mouse, joystick, Scanner, web cam, Output Devices- Monitors, Printers – Dot matrix, inkjet, laser, Multimedia- What is Multimedia, Text, Graphics, Animation, Audio, Images, Video; Multimedia Application in Education, Entertainment, Marketing.

UNIT - III OPERATING SYSTEM [04]

What is an operating system and basics of Windows-The User Interface Windows Setting

UNIT - IV SOFTWARE TYPES [04]

Names of common multimedia file formats, Computer Software- Relationship between Hardware and Software; System Software, Application Software, Compiler, names of some high level languages, free domain software.

UNIT - V COMPUTER NETWORK [04]

Networks-network types-web design –Applications-e-Commerce.

Total 20 Periods

COURSE OUTCOMES*: At the edd of the course, the student will be able to:*

Cos	***Course Outcomes***	**Cognitive Level (K1 to K6)**
CO1:	*Explain the concept of components, generations of computer.*	*K1*
CO2:	*Describe the concept of memory.*	*K1*
CO3:	*Demonstrate the concept of Operating system*	*K1*
CO4:	*Describe the Software types*	*K1*
CO5:	*Identify the purpose of computer Applications*	*K1*

References :

1 Anitha Goel, Computer Fundamentals, Pearson India, New Delhi, 2010.

2	Balagurusamy E, Fundamentals of computers, TMH, New Delhi, Second Edition, 2010.
3	Rajaraman V, Fundamentals of Computer, Prentice Hall of India Pvt. Ltd., New Delhi, Third Edition, 2012
4	Ram B, Computers Fundamentals Architecture and Organization, New Age International Publishers, New Delhi, Revised Edition, 2011.
5	Dhanpat Rai and Co, Fundamentals of Computer , Sumita Arora, New Delhi, Second Edition, 2013.
6	Sinha P.K., Computer Fundamentals, BPB Publications, New Delhi. Third Edition, 2010.

K.S.R. COLLEGE OF ENGINEERING (Autonomous)					**R 2024**
SEMESTER - I					
BC5002	**Problem Solving and Programming in C**	L	T	P	C
		2	0	0	0

***Prerequisite:** - . Introduction to Programming: Students should have a basic understanding of programming concepts, including data types, variables, control structures, functions, and arrays. Familiarity with programming fundamentals: Students should have experience with programming fundamentals, including loops, conditional statements, and functions. Basic understanding of computer systems: Students should have a basic understanding of computer systems, including hardware, software, and networking.*

Course Objectives:

1. Understand C programming fundamentals: Students will understand the basics of C programming, including data types, variables, control structures, functions, and arrays.
2. Write C programs: Students will learn to write efficient, readable, and well-documented C programs.
3. Understand memory management: Students will understand memory management concepts, including pointers, dynamic memory allocation, and memory deallocation.
4. Develop problem-solving skills: Students will develop problem-solving skills using C programming, including debugging and testing.
5. Understand C libraries and frameworks: Students will learn to use C libraries and frameworks, including the C Standard Library.

UNIT - I	**INTRODUCTION TO COMPUTER PROBLEM SOLVING**	**[04]**

Introduction – The Problem Solving aspect – Top down design – Implementation of algorithm – Program Verification – The efficiency of algorithms – The analysis of algorithms – Fundamental Algorithms

UNIT - II	**PROGRAMMING AND ALGORITHMS**	**[04]**

Programs and Programming – building blocks for simple programs -pseudo code representation – flow charts Programming Languages - compiler –Interpreter, Loader and Linker - Program execution – Classification of Programming Language – Structured Programming Concept – Illustrated Problems: Algorithm to check whether a given number is Armstrong number or not- Find factorial of a number.

UNIT - III	**BASICS OF ‘C’, INPUT / OUTPUT & CONTROL STATEMENTS**	**[04]**

Introduction- Identifier – Keywords - Variables – Constants – I/O Statements - Operators - Initialization – Expressions – Expression Evaluation – L values and R values – Type Conversion in C –Formatted input and output functions - Specifying Test Condition for Selection and Iteration- Conditional Execution - and Selection – Iteration and Repetitive

Execution- go to Statement – Nested Loops- Continue and break statements. Programs to be implemented:

1. Write a program to find whether the given year is leap year or Not? (Hint: not every centurion year is a leap. For example 1700, 1800 and 1900 is not a leap year)
2. Write a program to perform the Calculator operations, namely, addition, subtraction, multiplication, division and square of a number.

UNIT - IV	**ARRAYS, STRINGS, FUNCTIONS AND POINTERS**	**[04]**

Array – One dimensional Character Arrays- Multidimensional Arrays- Arrays of Strings – Two dimensional character array – functions - parameter passing mechanism scope – storage classes – recursion - comparing iteration and recursion- pointers – pointer operators - uses of pointers- arrays and pointers – pointers and strings - pointer indirection pointers to functions - Dynamic memory allocation.

1. Write a program in C to get the largest element of an array using the function.
2. Display all prime numbers between two intervals using functions.
3. Reverse a sentence using recursion.
4. Write a C program to concatenate two strings.
5. Find the frequency of a character in a string.

UNIT - V	**USER-DEFINED DATATYPES & FILES**	**[04]**

Structures – initialization - nested structures – structures and arrays – structures and pointers - union– type def and enumeration types - bit fields - File Management in C – Files and Streams – File handling functions – Sequential access file- Random access file – Command line arguments.

1. Write a C program to Store Student Information in Structure and Display it.
2. The annual examination is conducted for 10 students for five subjects.

3. Write a program to read the data and determine the following:
(a) Total marks obtained by each student.
(b) The highest marks in each subject and the marks of the student who secured it.
(c) The student who obtained the highest total marks.

Total 20 Periods

COURSE OUTCOMES*: At the edd of the course, the student will be able to:*

Cos	***Course Outcomes***	**Cognitive Level (K1 to K6)**
CO1:	Design a computational solution for a given problem.	*K1*
CO2:	Explain the break a problem into logical modules that can be solved (programmed).	*K1*
CO3:	Demonstrate transform a problem solution into programs involving programming constructs.	*K1*
CO4:	Illustrate programs using structures, strings, arrays, pointer and files for solving complex computational problem.	*K1*
CO5:	Explain introduce modularity using functions and pointers which permit ad hoc runtime polymorphism.	*K1*

Reference Books :

1	Deitel and Deitel, C How to Program, Pearson Education, Noida, Seventh Edition, 2013.
2	Byron S Gottfried, Programming with C, Schaums Outlines, Tata McGraw-Hill, New Delhi, Second Edition, 2011.
3	Brian W. Kernighan and Dennis M. Ritchie, The C programming Language, Pearson Education India, Second Edition, 2015,
4	Dromey R. G., How to solve it by Computer, Pearson Education, New Delhi, Fifth Edition, 2012.
5.	Kamthane, A.N., Programming with ANSI and Turbo C, Pearson Education, New Delhi, Third Edition, 2015.
6	Venugopal R, Sudeep R Prasad, Mastering CK, McGraw Hill Education Private Limited, New Delhi, India, Second Edition 2015.

<table>
<tr><td colspan="2"><b>K.S.R. COLLEGE OF ENGINEERING (Autonomous)</b></td><td colspan="4"><b>R 2024</b></td></tr>
<tr><td colspan="6"><b><u>SEMESTER - II</u></b></td></tr>
<tr><td rowspan="2"><b>BC5003</b></td><td rowspan="2"><b>Core Java Programming</b></td><td>L</td><td>T</td><td>P</td><td>C</td></tr>
<tr><td>2</td><td>0</td><td>0</td><td>0</td></tr>
</table>

Prerequisite: - *Introduction to Programming: Students should have a basic understanding of programming concepts, including data types, variables, control structures, functions, and arrays. 2. Familiarity with Object-Oriented Programming (OOP) concepts: Students should have a basic understanding of OOP concepts, including classes, objects, inheritance, polymorphism, and encapsulation. 3. Basic understanding of Java syntax: Students should have a basic understanding of Java syntax and semantics.*

Course Objectives:

1. Understand Java fundamentals: Students will understand the basics of Java programming, including data types, variables, operators, control structures, functions, and arrays.
2. Write Java programs: Students will learn to write efficient, readable, and well-documented Java programs.
3. Understand Object-Oriented Programming (OOP) concepts: Students will understand OOP concepts, including classes, objects, inheritance, polymorphism, and encapsulation.
4. Develop problem-solving skills: Students will develop problem-solving skills using Java programming, including debugging and testing.
5. Understand Java libraries and frameworks: Students will learn to use Java libraries and frameworks, including the Java Standard Library.

UNIT - I	**OVERVIEW OF OBJECT ORIENTED PROGRAMMING CONCEPTS**	**[04]**

Object Oriented Programming Concepts- Introduction- Methods –Method Overriding..

UNIT - II	**INTRODUCTION**	**[04]**

Java History-Applications-Lexical Issues- Data Types- Variables and Arrays Operators – Control Statements

UNIT - III	**PACKAGES**	**[04]**

Packages – Importing Packages- I/O Package – Interfaces – Exception Handling – Multithreaded Programming-String Operations. The Java I/O Classes– File – Byte Streams – The Character Streams – Serialization.

UNIT - IV	**JAVA PACKAGES: AWT, APPLET**	**[04]**

AWT Package: AWT Classes – Window Fundamentals – Working with Graphics– Working with Color – Working with Fonts – Applet Package: Applet Basics – Applet Architecture – Reading and Writing in Console –

UNIT - V	**JAVA SWING**	**[09]**

Java Swing-Applet-Icons and Labels-Text Fields-Buttons-Combo Box-Tabbed Panes-Scroll Panes-Trees-Tables-Exploring Swing.

Total (L: 45 T:0) = 45 Periods

COURSE OUTCOMES*: At the edd of the course, the student will be able to:*

Cos	***Course Outcomes***	**Cognitive Level (K1 to K6)**
CO1:	*Explain the basic object oriented programming concepts and know the origin of java programming.*	*K1*
CO2:	*Identify the different operations through single packages and understanding the String concepts.*	*K1*
CO3:	*Illustrate Learning the concept of java I/O packages.*	*K1*
CO4:	*Explain the concept of AWT package.*	*K1*
CO5:	*Describe the concept of swing*	*K1*

Reference Books :

1	Herbert Schildt, The Complete Reference JAVA, Tata McGraw Hill, New Delhi, ThirteenthEdition, 2017.
2	Joshua Bloch, Effective Java, Addison Wesley, New Delhi, Third Edition, 2016.
3	Herbert Schildt, The Complete Reference, Tata McGraw Hill, New Delhi, Eighth Edition, 2011.

4	Kogent, Java 6 Programming Black Book, Kogent Learning Solutions, New Delhi, Third Edition, 2011.
5.	Steven Holzner, Java2 (JDK 5 Edition) Programming, Tata McGraw Hill, New Delhi, Fourth Edition, 2012.
6	Balagurusamy E, Programming with Java, McGraw-Hill Education, New Delhi, Sixth Edition, 2019.

K.S.R. COLLEGE OF ENGINEERING (Autonomous)	R 2024

SEMESTER - II

		L	T	P	C
BC5004	**Software Engineering**	2	0	0	0

Prerequisite: Programming skills: Students should have proficiency in at least one programming language, such as Java, Python, C++, or JavaScript. Data structures and algorithms: Students should have a basic understanding of data structures (e.g., arrays, linked lists, stacks, queues) and algorithms (e.g., sorting, searching). Computer systems: Students should have a basic understanding of computer systems, including hardware, software, and networking.

Course Objectives:

1. Understand software engineering principles: Students will understand the fundamental principles of software engineering, including modularity, abstraction, and reuse.
2. Apply software development life cycle: Students will learn to apply the software development life cycle, including planning, analysis, design, implementation, testing, and maintenance.
3. Design and develop software systems: Students will learn to design and develop software systems using various programming languages and software development tools.
4. Test and validate software systems: Students will learn to test and validate software systems using various testing techniques, including unit testing, integration testing, and user acceptance testing.
5. Collaborate on software development projects: Students will learn to collaborate on software development projects, including team projects and version control systems.

UNIT - I	**INTRODUCTION**	**[04]**

Software Engineering Paradigms – Waterfall Life Cycle Model – Spiral Model – Prototype Model – Agile Process Model – Unified Process Model - Planning – Software Project Scheduling – SRS - Case Study: Project Plan and SRS

UNIT - II	**SOFTWARE DESIGN**	**[04]**

Designing Concepts - Abstraction – Modularity – Software Architecture – Cohesion – Coupling – Dataflow Oriented Design - Jackson System Development - Real time and Distributed System Design – Designing for Reuse —– Case Study : Design for any Application Oriented Project.

UNIT - III	**SOFTWARE TESTING AND MAINTENANCE**	**[04]**

Software Testing Fundamentals – Software Testing Strategies – Black Box Testing – White Box Testing – System Testing – Object Orientation Testing – State Based Testing – Testing Tools – Test Case Management – Types of Maintenance – Case Study: Testing Techniques

UNIT - IV	**SOFTWARE METRICS**	**[04]**

Scope – Classification of metrics – Measuring Process and Product attributes – Direct and Indirect measures – Cost Estimation - Reliability – Software Quality Assurance – Standards – Case Study for COCOMO model.

UNIT - V	**SCM & WEB ENGINEERING**	**[04]**

Need for SCM – Version Control – SCM process – Software Configuration Items – Taxonomy – Re Engineering – Reverse Engineering - Web Engineering - CASE Repository – Features.

Total 20 Periods

COURSE OUTCOMES*: At the edd of the course, the student will be able to:*

Cos	***Course Outcomes***	**Cognitive Level (K1 to K6)**
CO1:	Explain the problem domain to choose process models and to develop SRS	*K1*
CO2:	Summarize the model software projects using appropriate design notations	*K1*
CO3:	Show the measure the product and process performance using various metrics	*K1*
CO4:	Evaluate the system with various testing techniques and strategies	*K1*
CO5:	Analyze, design, verify, validate, implement, and maintain software systems.	*K1*

Reference Books :

1	Roger S. Pressman, Software Engineering: A Practitioner Approach, Tata McGraw – Hill International Edition, New Delhi, Seventh Edition, 2017

2	Richard Fairley, Software Engineering Concepts, Tata McGraw Hill Edition, New Delhi, Third Edition, 2018.
3	Ali Behforroz, Frederick J.Hudson, Software Engineering Fundamentals, Oxford Indian Reprint, New Delhi, First Edition, 2012
4	Sommerville, Software Engineering, Pearson, New Delhi, Tenth Edition, 2016.
5	PankajJalote, An Integrated approach to Software Engineering, Narosa Publications, New Delhi, Third Edition, 2011
6	David Farley, Modern Software Engineering: Doing What Works to Build, Tata McGraw Hil, New Delhi, First Edition, 2021