KSR College of Engineering

NAAC ACCREDITED A++ | NBA ACCREDITED PROGRAMMES

AN AUTONOMOUS INSTITUTION

M.E. - COMPUTER SCIENCE AND ENGINEERING

Curriculum &
Syllabus for Semester I and II

REGULATIONS 2024

(Academic Year 2024-25 Onwards)

K.S.R. COLLEGE OF ENGINEERING: TIRUCHENGODE - 637 215

(Autonomous)

DEPARTMENT OF COMPUTER SCIENCE AND ENIGNEERING

M.E. - Computer Science and Engineering

(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

IM1	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 mutually beneficial partnership with global industries and Institutions through knowledge sharing, collaborative research and innovation.

Vision of the Department / Programme: (Computer Science and Engineering)

DV	To create ever green professionals for software industry, academicians for knowledge cultivation and researchers for contemporary society modernization.

Mission of the Department / Programme: (Computer Science and Engineering)

DM 1	To produce proficient design, code and system engineers for software development.
DM 2	To keep updated contemporary technology and fore coming challenges for welfare of the society.

Programme Educational Objectives (PEOs): (Computer Science and Engineering)

The graduates of the programme will be able to	
PEO 1	**Engineering knowledge:** Apply the necessary mathematical tools and fundamental & advanced knowledge of computer science & engineering.
PEO 2	**Development of solutions:** Develop computer/software/network systems understanding the importance of social, business, technical, environmental, and human context in which the systems would work.
PEO 3	**Individual and Teamwork:** Contribute effectively as a team member/leader, using common tools and environment, in computer science and engineering projects, research, or education.

Programme Outcomes (POs) of B.E. - Computer Science and Engineering

PO1	**M.E Computer Science and Engineering graduates will be able to attain:** An ability to independently carry out research /investigation and development work to solve practical problems.
PO2	An ability to write and present a substantial technical report/document.
PO3	Students should be able to demonstrate a degree of mastery over the area as per the specialization of the program. The mastery should be at a level higher than the requirements in the appropriate bachelor program.
PSO1	**Computer System Design:** Apply the knowledge of computer system design principles in building system software and hardware components.
PSO2	**Solve Computational Problems:** Apply the theoretical foundations of computer science in modeling and developing solutions to the real-world problems.

KSR College of Engineering	**K. S. R COLLEGE OF ENGINEERING** **An Autonomous Institution** **Approved by AICTE and Affiliated to Anna University, Chennai** **Accredited by NAAC ('A++' Grade)**	**Curriculum** **PG** **R - 2024**
Department	**Department of Computer Science and Engineering**	
Programme	**M.E. Computer Science and Engineering**	

SEMESTER I

S. No.	Course Code	Course Title	Category	Periods / Week				Credit	Max. Marks		
				L	T	P	Tot		CA	ES	Tot
Induction Programme			-	-	-	-	-	-	-	-	-
THEORY COURSES											
1	MA24T16	Operations Research	FC	3	0	0	3	**3**	40	60	100
2	CS24T16	Advanced Data Structures and Algorithms	PCC	3	0	0	3	**3**	40	60	100
3	CS24T17	Database Practices	PCC	3	0	0	3	**3**	40	60	100
4	CS24T18	Network Technologies	PCC	3	0	0	3	**3**	40	60	100
5		Professional Elective - I	PEC	3	0	0	3	**3**	40	60	100
6		Professional Elective - II	PEC	3	0	0	3	**3**	40	60	100
LABORATORY COURSES											
7	CS24P16	Advanced Data Structures and Algorithms Laboratory	PCC	0	0	3	3	**2**	60	40	100
8	CS24P11	Database Practices Laboratory	PCC	0	0	3	3	**2**	60	40	100
			TOTAL	**18**	**0**	**6**	**24**	**22**	**800**		

SEMESTER II

S. No.	Course Code	Course Title	Category	Periods / Week				Credit	Max. Marks		
				L	T	P	Tot		CA	ES	Tot
THEORY COURSES											
1	RM24T09	Research Methodology and IPR	RMC	3	0	0	3	**3**	40	60	100
2	BD24T26	Big Data Mining and Analytics	PCC	3	0	0	3	**3**	40	60	100
3	CS24T27	Cloud Computing	PCC	3	0	0	3	**3**	40	60	100
4	BD24E08	Internet of Things	PCC	3	0	0	3	**3**	40	60	100
5		Professional Elective - III	PEC	3	0	0	3	**3**	40	60	100
6		Professional Elective - IV	PEC	3	0	0	3	**3**	40	60	100
LABORATORY COURSES											
7	BD24P26	Big Data Mining and Analytics Laboratory	PCC	0	0	3	3	**2**	60	40	100
8	CS24P26	Cloud Computing Laboratory	PCC	0	0	3	3	**2**	60	40	100
9	CS24P21	Technical Presentation	EEC	0	0	3	3	**2**	60	40	100
		TOTAL		**18**	**0**	**9**	**27**	**24**	**900**		

SEMESTER III											
S. No.	**Course Code**	**Course Title**	**Category**	**Periods / Week**				**Credit**	**Max. Marks**		
				L	**T**	**P**	**Tot**		**CA**	**ES**	**Tot**
THEORY COURSES											
1	CS24T31	Security Practices	PCC	3	0	0	3	**3**	40	60	100
2	CS24T36	Soft Computing	PCC	3	0	0	3	**3**	40	60	100
3		Professional Elective - V	PEC	3	0	0	3	**3**	40	60	100
4		Open Elective - I	OEC	3	0	0	3	**3**	40	60	100
5		Audit courses	AC	2	0	0	2	**0**	100	-	100
LABORATORY COURSES											
6	CS24P31	Project Phase – I	EEC	0	0	12	12	**6**	60	40	100
TOTAL				**14**	**0**	**12**	**26**	**18**	**600**		
SEMESTER IV											
S. No.	**Course Code**	**Course Title**	**Category**	**Periods / Week**				**Credit**	**Max. Marks**		
				L	**T**	**P**	**Tot**		**CA**	**ES**	**Tot**
LABORATORY COURSES											
1	CS24P41	Project Phase – II	EEC	0	0	24	24	**12**	60	40	100
TOTAL				**0**	**0**	**24**	**24**	**12**	**100**		
TOTAL CREDITS								**76**			

TOTAL NUMBER OF CREDITS TO BE EARNED FOR AWARD OF THE DEGREE = 76
Note: FC - Foundation Courses, PCC - Professional Core Courses, RMC - Research Methodology Courses, PEC - Professional Elective Courses, EEC - Employability Enhancement Courses and AC - Audit Courses.

FOUNDATION COURSES (FC)											
S. No.	**Course Code**	**Course Title**	**Semester**	**Periods / Week**				**Credit**	**Max. Marks**		
				L	**T**	**P**	**Tot**		**CA**	**ES**	**Tot**
1	MA24T16	Operations Research	I	3	0	0	3	**3**	40	60	100
TOTAL				**3**	**0**	**0**	**3**	**3**			
PROFESSIONAL CORE COURSES (PCC)											
S. No.	**Course Code**	**Course Title**	**Semester**	**Periods / Week**				**Credit**	**Max. Marks**		
				L	**T**	**P**	**Tot**		**CA**	**ES**	**Tot**
1	CS24T16	Advanced Data Structures and Algorithms	I	3	0	0	3	**3**	40	60	100
2	CS24T17	Database Practices	I	3	0	0	3	**3**	40	60	100
3	CS24T18	Network Technologies	I	3	0	0	3	**3**	40	60	100

4	CS24P16	Advanced Data Structures and Algorithms Laboratory	I	0	0	3	3	**2**	60	40	100
5	CS24P11	Database Practices Laboratory	I	0	0	3	3	**2**	60	40	100
6	BD24T26	Big Data Mining and Analytics	II	3	0	0	3	**3**	40	60	100
7	CS24T27	Cloud Computing	II	3	0	0	3	**3**	40	60	100
8	BD24E08	Internet of Things	II	3	0	0	3	**3**	40	60	100
9	BD24P26	Big Data Mining and Analytics Laboratory	II	0	0	3	3	**2**	60	40	100
10	CS24P26	Cloud Computing Laboratory	II	0	0	3	3	**2**	60	40	100
11	CS24T31	Security Practices	III	3	0	0	3	**3**	40	60	100
12	CS24T36	Soft Computing	III	3	0	0	3	**3**	40	60	100
TOTAL				**24**	**0**	**12**	**36**	**32**			

EMPLOYABILITY ENHANCEMENT COURSES (EEC)

S. No.	Course Code	Course Title	Semester	Periods / Week				Credit	Max. Marks		
				L	T	P	Tot		CA	ES	Tot
1	CS24P21	Technical Presentation	II	0	0	3	3	**2**	60	40	100
2	CS24P31	Project Phase – I	III	0	0	12	12	**6**	60	40	100
3	CS24P41	Project Phase – II	IV	0	0	24	24	**12**	60	40	100
TOTAL				**0**	**0**	**39**	**39**	**20**			

RESEARCH METHODOLOGY COURSES(REC)

S. No.	Course Code	Course Title	Semester	Periods / Week				Credit	Max. Marks		
				L	T	P	Tot		CA	ES	Tot
1	RM24T09	Research Methodology and IPR	II	3	0	0	3	**3**	40	60	100
TOTAL				**3**	**0**	**0**	**3**	**3**			

PROFESSIONAL ELECTIVE COURSES (PEC)
PROFESSIONAL ELECTIVES – I and II (SEMESTER – I)

S. No.	Course Code	Course Title	Semester	Periods / Week				Credit	Max. Marks		
				L	T	P	Tot		CA	ES	Tot
1	CS24E01	Data Mining Techniques	I	3	0	0	3	**3**	40	60	100
2	CS24E02	Advanced Operating System	I	3	0	0	3	**3**	40	60	100
3	CS24E03	Mobile and Pervasive Computing	I	3	0	0	3	**3**	40	60	100
4	BD24T16	Foundations of Data Science	I	3	0	0	3	**3**	40	60	100
5	BD24E04	Agile Methodologies	I	3	0	0	3	**3**	40	60	100
6	CS24E04	Object Oriented Software Engineering	I	3	0	0	3	**3**	40	60	100

7	CS24E05	Wireless Sensor Networks	I	3	0	0	3	**3**	40	60	100
8	CS24E06	Multicore Architectures	I	3	0	0	3	**3**	40	60	100
9	CS24E07	Human Computer Interaction	I	3	0	0	3	**3**	40	60	100
10	BD24E05	Web Services and API Design	I	3	0	0	3	**3**	40	60	100

PROFESSIONAL ELECTIVE COURSES (PEC)
PROFESSIONAL ELECTIVES – III and IV (SEMESTER – II)

S. No.	Course Code	Course Title	Semester	Periods / Week				Credit	Max. Marks		
				L	T	P	Tot		CA	ES	Tot
1	BD24T27	Machine Learning Techniques	II	3	0	0	3	**3**	40	60	100
2	CS24E08	Software Quality Assurance	II	3	0	0	3	**3**	40	60	100
3	CS24E09	Full Stack Web Application Development	II	3	0	0	3	**3**	40	60	100
4	CS24E10	Deep Learning	II	3	0	0	3	**3**	40	60	100
5	CS24E11	Natural Language Processing	II	3	0	0	3	**3**	40	60	100
6	BD24E09	Blockchain Technologies	II	3	0	0	3	**3**	40	60	100
7	BD24E10	Cyber Physical Systems	II	3	0	0	3	**3**	40	60	100
8	CS24E12	GPU Computing	II	3	0	0	3	**3**	40	60	100
9	CS24E13	Quantum Computing	II	3	0	0	3	**3**	40	60	100
10	BD24E12	Information Retrieval Techniques	II	3	0	0	3	**3**	40	60	100

PROFESSIONAL ELECTIVE COURSES (PEC)
PROFESSIONAL ELECTIVES – V (SEMESTER – III)

S. No.	Course Code	Course Title	Semester	Periods / Week				Credit	Max. Marks		
				L	T	P	Tot		CA	ES	Tot
1	BD24E14	DevOps and Microservices	III	3	0	0	3	**3**	40	60	100
2	CS24E14	Augmented Reality and Virtual Reality	III	3	0	0	3	**3**	40	60	100
3	CS24E15	Software Industrialization	III	3	0	0	3	**3**	40	60	100
4	CS24E16	Digital Image Processing	III	3	0	0	3	**3**	40	60	100
5	BD24E17	Social Network Analysis	III	3	0	0	3	**3**	40	60	100

OPEN ELECTIVE OFFERED BY OTHER DEPARTMENT

1	BD24O03	Big Data Visualization	III	3	0	0	3	3	40	60	100
2	PE24O03	Renewable Energy Technology	III	3	0	0	3	3	40	60	100
3	ET24O01	Embedded Systems	III	3	0	0	3	3	40	60	100
4	CN24O03	Stress management	III	3	0	0	3	3	40	60	100

5	ST24O01	Principles of Sustainable development	III	3	0	0	3	3	40	60	100
6	CU24O01	Principles of Multimedia	III	3	0	0	3	3	40	60	100
7	IS24O01	Industrial Safety Engineering	III	3	0	0	3	3	40	60	100
8	IS24O03	Food and Bio-safety	III	3	0	0	3	3	40	60	100
OPEN ELECTIVE OFFERED TO OTHER DEPARTMENT											
9	CS24O01	Machine learning and Deep Learning	III	3	0	0	3	**3**	40	60	100
10	CS24O02	Blockchain and Crypto Currency	III	3	0	0	3	**3**	40	60	100
11	CS24O03	Multimedia Technologies	III	3	0	0	3	**3**	40	60	100

AUDIT COURSES (SEMESTER – III)

S. No.	Course Code	Course Title	Semester	Periods / Week				Credit	Max. Marks		
				L	T	P	Tot		CA	ES	Tot
1	CS24A01	Disaster Management	III	2	0	0	2	**0**	100	-	100
2	CS24A02	Value Education	III	2	0	0	2	**0**	100	-	100
3	CS24A03	Constitution of India	III	2	0	0	2	**0**	100	-	100
4	CS24A04	Indian Knowledge System	III	2	0	0	2	**0**	100	-	100

Summary

Name of the Programme: M.E Computer science and Engineering

CATEGORY	I	II	III	IV	TOTAL CREDITS	%
FC	3	-	-	-	**3**	**3.94**
PCC	13	13	6	-	**32**	**42.10**
REC	-	3	-	-	**3**	**3.94**
PEC	6	6	3	-	**15**	**19.73**
OEC	-	-	3	-	**3**	**3.94**
EEC	-	2	6	12	**20**	**26.31**
AC	-	-	✓	-	**-**	**-**
Total	**22**	**24**	**18**	**12**	**76**	**100**

MA24T16	OPERATIONS RESEARCH	Category	L	T	P	C
		FC	3	0	0	3

(Common to M. E CSE, M. E BDA and M.Tech IT)

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, Queuing model.

OBJECTIVES:

1. To determine the most effective way to allocate the best value of linear programming such as profit or loss based on decision variables.
2. To analyze the most effective way to minimize the total transportation cost and to find the optimal way to assign a set of tasks.
3. To determine the optimal quantity of inventory to hold the balancing between excess and shortage and improve optimal efficiency and reduce waste.
4. To develop the ability to analyze the basic components and behavior of queuing systems
5. To facilitate learners about the PERT/CPM models to identify shortest path, Network design, Project Scheduling.

UNIT – I	LINEAR PROGRAMMING	(9)

Formation of LPP – Graphical method – Simplex method – Big M Method – Dual simplex method.

UNIT – II	TRANSPORTATION AND ASSIGNMENT PROBLEMS	(9)

Transportation Models (Minimizing and Maximizing Problems) – Balanced and unbalanced Problems – Initial Basic feasible solution by North West Corner Rule, least cost and Vogel's approximation methods – Optimum solution by MODI Method – Assignment Models (Minimizing and Maximizing Problems) – Hungarian method - Balanced and Unbalanced Problems.

UNIT – III	INVENTORY MODELS	(9)

Types of Inventory – Deterministic inventory models: Purchasing problem with no shortage and with shortages – Production problem with and without shortages – Purchase problem with price breaks – Probabilistic inventory model (excluding proof).

UNIT – IV	QUEUING MODELS	(9)

Characteristics of Queuing Models – Kendall's notations - Little's formula – (M/M/1): (∞/FIFO) Single Server with infinite capacity – (M/M/C):(∞/FIFO) Multi Server with infinite capacity – (M/M/1) : (N/FIFO) Single Server with finite capacity – (M/M/C) : (N/FIFO) Multi server with finite capacity

UNIT – V	PERT/CPM	(9)

Network Construction – Critical Path Method – Computation of earliest start time, latest start time, Total, free and independent float time – PERT Analysis – Computation of optimistic, most likely Pessimistic and expected time.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
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	Evaluate the inventory model using EOQ and EBQ with and without shortage.	Apply
CO4	Analyze and interpret the key features of various queuing systems	Analyze
CO5	Apply and evaluate the concepts of network model	Analyze

TEXT BOOKS:

1. Taha H.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. Hira and 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	PSO1	PSO2
CO1	3	-	-	2	-
CO2	3	-	-	2	-
CO3	3	-	-	2	-
CO4	3	-	-	2	-
CO5	3	-	-	2	-
Avg.	**3**	**-**	**-**	**2**	**-**

1-low, 2-medium, 3-high

CS24T16	ADVANCED DATA STRUCTURES AND ALGORITHMS	Category	L	T	P	C
		PCC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Familiarity with basic data structures like arrays, linked lists, stacks, and queues is essential for understanding advanced topics. This foundational knowledge helps in grasping how more complex data structures and algorithms build upon these basics to address intricate problems.

OBJECTIVES:

6. To introduce the fundamental concepts of algorithm efficiency and complexity analysis.
7. To explore advanced tree and heap data structures for efficient data management
8. To teach graph algorithms for traversal, shortest paths, and minimum spanning trees.
9. To learn the skills in applying dynamic programming and greedy strategies for optimization.
10. To study the principles of NP-completeness, NP-hardness, and approximation algorithms.

UNIT – I	ROLE OF ALGORITHMS IN COMPUTING	(9)

Algorithms – Algorithms as a Technology – Time and Space complexity of algorithms – Asymptotic analysis – Average and Worst-case analysis – Asymptotic notation – Importance of efficient algorithms – Program performance measurement – Recurrences: The Substitution Method – The Recursion– Tree Method.

UNIT – II	HIERARCHICAL DATA STRUCTURES	(9)

Binary Search Trees – Red Black trees – B-Trees – B+ Trees – AVL Tree – Multi-way Search Trees – Heap: Heap Operations – Min/Max heaps – Fibonacci Heaps: Structure – Mergeable-heap operations– Decreasing a key and deleting a node– Bounding the maximum degree.

UNIT – III	GRAPH	(9)

Graph: Representations of Graphs – Breadth-First Search – Depth-First Search – Topological Sort – Minimum Spanning Trees: Growing a Minimum Spanning Tree – Kruskal and Prim's – Single-Source Shortest Paths: The Bellman-Ford algorithm – Dijkstra's Algorithm – All-Pairs Shortest Paths: Shortest Paths and Matrix Multiplication – The Floyd -Warshall Algorithm.

UNIT – IV	ALGORITHM DESIGN TECHNIQUES	(9)

Dynamic Programming: Matrix-Chain Multiplication – Optimal binary search trees – Elements of Dynamic Programming – Longest Common Subsequence – Greedy Algorithms: An Activity – Selection Problem – Elements of the Greedy Strategy – Huffman Codes and Trees.

UNIT – V	NP COMPLETE AND NP HARD	(9)

NP Completeness: Polynomial Time – Polynomial Time Verification – NP Completeness and Reducibility – Proof of NP hardness and NP completeness – Approximation algorithms – Randomized Algorithms.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Comprehend fundamental concepts of algorithm efficiency and apply complexity analysis methods.	**Understand**
CO2	Recognize and implement advanced hierarchical data structures for effective data management.	**Understand**
CO3	Apply graph algorithms to solve shortest paths and spanning trees and analyze their results.	**Apply**
CO4	Identify dynamic programming and greedy strategies and synthesize these techniques for optimization.	**Understand**
CO5	Analyze NP-complete problems and develop solutions using approximation and randomized algorithms.	**Apply**

REFERENCES:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", MIT Press, London, 4 th Edition, 2022.
2. Reema Thareja, "Data Structures Using C", Oxford University Press, England, 3rd Edition, 2023.
3. Weiss, "Data Structures and Algorithm Analysis in C", Pearson Education, India, 2nd Edition, 2015.
4. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, "Data Structures and Algorithms", Pearson Education, India, Reprint 2006.
5. S.Sridhar, "Design and Analysis of Algorithms", Oxford University Press, England, 1st Edition, 2014.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	-	3	1	3
CO2	3	-	3	1	3
CO3	3	-	3	1	3
CO4	3	-	3	1	3
CO5	3	-	3	1	3
Avg.	**3**	**-**	**3**	**1**	**3**

1-low, 2-medium, 3-high

CS24T17	DATABASE PRACTICES	Category	L	T	P	C
		PCC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE
Students should have a foundation in SQL, relational algebra and basic database design principles. Knowledge of distributed databases, XML, and introductory NoSQL systems is also needed.

OBJECTIVES:
1. To explore fundamental concepts of the relational model and SQL.
2. To learn database design strategies using E-R modelling and normalization.
3. To study distributed databases, active databases, and open database connectivity.
4. To know XML data models and management methods within databases.
5. To examine NoSQL databases and big data storage technologies like Hadoop and Map Reduce.

UNIT – I	RELATIONAL DATA MODEL	(9)

Introduction to the Relational Model – Relational Algebra – Introduction to SQL – Intermediate SQL – Advanced SQL.

UNIT – II	DATABASE DESIGN	(9)

Database Design Using the E-R Model: Overview of the Design Process – The Entity-Relationship Model – Complex Attributes – Mapping Cardinalities – Removing Redundant Attributes in Entity Sets – Reducing E-R Diagrams to Relational Schemas – Extended E-R Features – Entity-Relationship Design Issues – Alternative Notations for Modeling Data – Relational Database Design: Features of Good Relational Designs – Functional Dependencies – Non-loss Decomposition – First, Second and Third Normal Forms, Dependency Preservation – Boyce/Codd Normal Form – Multi-valued Dependencies and Fourth Normal Form – Join Dependencies and Fifth Normal Form.

UNIT – III	DISTRIBUTED DATABASES, ACTIVE DATABASES AND OPEN DATABASE CONNECTIVITY	(9)

Distributed Database Architecture – Distributed Data Storage – Distributed Query Processing – Distributed Transaction Processing – Active Database Concepts and Triggers – Design and Implementation Issues for Active Databases – Open Database Connectivity.

UNIT – IV	XML DATABASES	(9)

Structured, Semi structured and Unstructured Data – XML Hierarchical Data Model – XML Documents – Document Type Definition – XML Schema – Storing and Extracting XML Documents from Databases – XML Languages – Extracting XML Documents from Relational Databases – XML/SQL: SQL Functions for Creating XML Data.

UNIT – V	NOSQL DATABASES AND BIG DATA STORAGE SYSTEMS	(9)

Introduction to NOSQL Systems – CAP Theorem – Document-Based NoSQL Systems and MongoDB – NoSQL Key-Value Stores – Dynamo DB Overview – Voldemort Key-Value Distributed Data Store – Column-Based or Wide Column NoSQL Systems – NoSQL Graph Databases and Neo4j – Big Data – MapReduce – Hadoop – YARN.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Interpret and utilize the principles of the relational model and SQL effectively.	**Understand**
CO2	Comprehend and apply E-R modeling and normalization.	**Apply**
CO3	Familiarize with and analyze distributed and active databases and connectivity.	**Understand**
CO4	Recognize XML data models and apply XML management methods.	**Apply**
CO5	Examine NoSQL databases and evaluate big data technologies.	**Apply**

REFERENCES:

1. Abraham Silberschatz, Henry F. Korth and S. Sudharshan, "Database System Concepts", Tata McGraw Hill, New Delhi, 7th Edition, 2019.
2. Ramez Elmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", Pearson Education, New Delhi, 7th Edition, 2016.
3. S.K.Singh, "Database Systems Concepts, Design and Applications", Pearson Education, New Delhi, 2nd Edition, 2011.
4. Harrison, Guy, "Next Generation Databases, NoSQL and Big Data", Apress publishers, Pune, 1st Edition, 2015.
5. Thomas Cannolly and Carolyn Begg, "Database Systems, A Practical Approach to Design, Implementation and Management", Pearson Education, New Delhi, 6th Edition, 2015.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	-	1	3	1
CO2	3	-	1	3	1
CO3	3	-	1	3	1
CO4	3	-	1	3	1
CO5	3		1	3	1
Avg.	**3**	**-**	**1**	**3**	**1**

1-low, 2-medium, 3-high

CS24T18	NETWORK TECHNOLOGIES	Category	L	T	P	C
		PCC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Students should have a basic understanding of networking terminology, OSI model principles and network devices. Familiarity with IP addressing, wireless networks, and virtualization concepts is also needed. Basic knowledge of programming and network security.

OBJECTIVES:

1. To introduce fundamental networking concepts, devices and terminology.
2. To explore different wireless network technologies and standards.
3. To review sophisticated mobile network technologies and protocols
4. To learn the principles and architecture of Software Defined Networking (SDN).
5. To study the design and advantages of Network Functions Virtualization (NFV)

UNIT – I	**NETWORKING CONCEPTS**	**(9)**

Peer To Peer Vs Client-Server Networks – Network Devices – Network Terminology – Network Speeds – Network throughput delay – OSI Model Packets – Frames – Headers – Collision and Broadcast Domains – LAN Vs WAN Network Adapter – Hub – Switch – Router – Firewall – IP addressing.

– **UNIT – II**	**WIRELESS NETWORKS**	**(9)**

Wireless access techniques – IEEE 802.11a, 802.11g, 802.11e, 802.11n/ac/ax/ay/ba/be, QoS – Bluetooth – Protocol Stack – Security – Profiles – zigbee

UNIT – III	**MOBILE DATA NETWORKS**	**(9)**

4G Networks and Composite Radio Environment – Protocol Boosters – Hybrid 4G Wireless Networks Protocols – Green Wireless Networks – Physical Layer and Multiple Access – Channel Modelling for 4G – Concepts of 5G – Channel access – Air Interface – Cognitive Radio – Spectrum Management – C-RAN Architecture – Vehicular Communications – Protocol – Network Slicing – MIMO – mmWave – Introduction to 6G.

UNIT – IV	**SOFTWARE DEFINED NETWORKS**	**(9)**

SDN Architecture – Characteristics of Software – Defined Networking – SDN and NFV Related Standards – SDN Data Plane – Data Plane Functions – Data Plane Protocols – OpenFlow Logical Network Device – Flow Table Structure – Flow Table Pipeline – Use of Multiple Tables – Group Table – OpenFlow Protocol – SDN Control Plane Architecture – Control Plane Functions – Southbound Interface – Northbound Interface – Routing – ITU-T Model.

UNIT – V	**NETWORK FUNCTIONS VIRTUALIZATION**	**(9)**

Motivation – Virtual Machines – NFV Benefits and Requirements – Architecture – NFV Infrastructure – Virtualized Network Functions – NFV Management and Orchestration – NFV Use Cases – NFV and SDN – Network virtualization – VLAN and VPN.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Identify and apply fundamental networking concepts and terminology.	**Apply**
CO2	Familiarize with and assess various wireless network technologies and standards.	**Understand**
CO3	Examine and critique advanced mobile network technologies and protocols.	**Apply**
CO4	Comprehend and implement the principles and architecture of Software Defined Networking (SDN).	**Apply**
CO5	Realize and appraise the design and advantages of Network Functions Virtualization (NFV).	**Understand**

REFERENCES:

1. James Bernstein, "Networking made Easy", Independently Published, 1st Edition, 2018.
2. Houda Labiod, Costantino de Santis, Hossam Afifi, "Wi-Fi, Bluetooth, Zigbee and WiMax", Springer, Netherlands, 1st Edition, 2007.
3. Erik Dahlman, Stefan Parkvall, Johan Skold, "4G: LTE/LTE-Advanced for Mobile Broadband", Academic Press, Cambridge, 1st Edition, 2013.
4. Saad Z. Asif, "5G Mobile Communications Concepts and Technologies", CRC press, Florida, 1st Edition, 2019.
5. William Stallings, "Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud", Pearson Education, 1st Edition, 2016.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	1	1	2	2
CO2	3	1	1	2	2
CO3	3	1	1	2	2
CO4	3	1	1	2	2
CO5	3	1	1	2	2
Avg.	**3**	**1**	**1**	**2**	**2**

1-low, 2-medium, 3-high

CS24P16	ADVANCED DATA STRUCTURES AND ALGORITHMS LABORATORY	Category	L	T	P	C
		PCC	0	0	3	2

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Students should have a basic understanding of programming, data structures and fundamental algorithms. Familiarity with mathematical concepts related to recursion and algorithm complexity, along with strong problem-solving skills

OBJECTIVES:

1. To gain proficiency in developing and applying recursive functions for various computational problems.
2. To achieve competence in implementing and evaluating different sorting algorithms for effective data management.
3. To acquire skills in creating and managing various tree data structures for efficient data organization and retrieval.
4. To become adept at applying algorithms for solving graph-related problems, such as finding shortest paths and constructing minimum spanning trees.
5. To develop the ability to implement and analyze algorithms for complex optimization problems and coding challenges.

List of Exercise/Experiments:

1. Develop recursive methods for tree traversal (In-order, Pre-order, Post-order) and for calculating Fibonacci numbers.
2. Generate solutions for Merge Sort and Quick Sort algorithms.
3. Construct a Binary Search Tree (BST) with essential operations like insertion, deletion, and search.
4. Design and build a Red-Black Tree, ensuring it maintains its balancing properties.
5. Construct a Heap (Min-Heap or Max-Heap) and perform standard heap operations.
6. Assemble a Fibonacci Heap and utilize it for efficient priority queue operations.
7. Develop Prim’s algorithm to determine the Minimum Spanning Tree of a graph.
8. Implementation of minimum cost spanning tree using Kruskal’s algorithm.
9. Design Dijkstra's algorithm and Bellman-Ford algorithm to compute the shortest paths from a single source in a graph.
10. Write a program to compute the shortest path from a single source to all other vertices in a given graph.
11. Develop an algorithm to solve the Matrix Chain Multiplication problem, optimizing the sequence of matrix multiplications.
12. Design the Activity Selection problem using a greedy strategy and Huffman Coding Implementation.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Develop and execute recursive algorithms for problem-solving tasks.	**Apply**
CO2	Apply and compare sorting techniques to evaluate their performance and efficiency.	**Apply**
CO3	Build and work with various tree structures to organize and access data effectively.	**Apply**
CO4	Implement graph algorithms to solve problems related to shortest paths and minimum spanning trees.	**Apply**
CO5	Develop and apply algorithms for complex optimization and coding challenges.	**Apply**

REFERENCES:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", MIT Press, London, 4 th Edition, 2022.
2. Reema Thareja, "Data Structures Using C", Oxford University Press, England, 3 rd Edition, 2023.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	**3**	**-**	**3**	**3**	**2**
CO2	**3**	**-**	**3**	**3**	**2**
CO3	**3**	**-**	**3**	**3**	**2**
CO4	**3**	**-**	**3**	**3**	**2**
CO5	**3**	**-**	**3**	**3**	**2**
Avg.	**3**	**-**	**3**	**3**	**2**

1-low, 2-medium, 3-high

CS24P11	DATABASE PRACTICES LABORATORY	Category	L	T	P	C
		PCC	0	0	3	2

PREREQUISITE

Students should have basic knowledge of SQL and relational databases, including table creation and data manipulation. Familiarity with programming languages for database access and understanding XML and NoSQL databases.

OBJECTIVES:

1. To practice creating, altering, and managing database tables and constraints.
2. To perform data manipulation tasks such as selection, insertion, updating, and deletion.
3. To execute advanced queries involving set operations, aggregate functions, string operations, and joins.
4. To create and manage database views and triggers for real-time applications.
5. To integrate and manage XML documents in relational databases and query various NoSQL databases.

List of Exercise/Experiments:

1. Develop and execute SQL Data Definition Commands to perform the following for real-time applications:
    i. Create, Alter, Rename, Truncate and Drop Table
    ii. Enforce Primary Key, Foreign Key, Check, Unique and Not Null Constraint
2. Formulate and execute SQL Data Manipulation Language statements to perform Select, Insert, Update, and Delete operations in the real-time database.
3. Construct and execute database queries involving Set operations, Aggregate functions, String operations, and Joins in the real-time database.
4. Design and implement SQL Views for real-time applications.
5. Create and apply Triggers for real-time databases.
6. Develop and implement solutions to access a Relational Database using Java, Python, PHP, or R for a real-time application.
7. Generate XML Documents, Document Type Definitions, and XML Schemas for real-time applications.
8. Execute operations to store XML documents as text in a Relational Database for real-time applications.
9. Extract XML Documents from Relational Databases for real-time applications.
10. Compose and run queries to access databases created using MongoDB, DynamoDB, Voldemort Key-Value Distributed Data Store, HBase, and Neo4j.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Execute SQL DDL commands to create and modify tables and enforce constraints.	**Apply**
CO2	Perform data manipulation tasks using SQL DML commands for inserting, updating, and deleting records.	**Apply**
CO3	Apply advanced SQL querying techniques, including set operations, aggregate functions, and joins.	**Apply**
CO4	Design and implement SQL views and triggers for managing and automating database operations.	**Apply**
CO5	Handle XML data within relational databases and perform queries on various NoSQL databases.	**Apply**

REFERENCES:

1. Abraham Silberschatz, Henry F. Korth and S. Sudharshan, "Database System Concepts", Tata McGraw Hill, New Delhi, 7th Edition, 2019.
2. Ramez Elmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", Pearson Education, New Delhi, 7th Edition, 2016.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	**3**	**1**	**2**	**2**	**1**
CO2	**3**	**1**	**2**	**2**	**1**
CO3	**3**	**1**	**2**	**2**	**1**
CO4	**3**	**1**	**2**	**2**	**1**
CO5	**3**	**1**	**2**	**2**	**1**
Avg.	**3**	**1**	**2**	**2**	**1**

1-low, 2-medium, 3-high

RM24T09	RESEARCH METHODOLOGY AND IPR	Category	L	T	P	C
		RMC	3	0	0	3

(Common to PED, EST, CSE, BDA, CAD CAM, ISE)

PREREQUISITE: A basic understanding of academic writing and critical thinking skills to analyze research literature, familiarity with fundamental statistical concepts for data analysis, and a strong grasp of core subject knowledge relevant to the student's field. Additionally, a general awareness of legal principles related to intellectual property, ethical research practices, and innovation trends will enhance the student's ability to engage with both research methodologies and IPR concepts.

OBJECTIVES:

- To equip students with the ability to design and conduct rigorous research, employing appropriate methodologies, and critically analyzing results.
- To foster the ability to critically evaluate academic literature, identify research gaps, and formulate research questions.
- To enable students to effectively communicate research findings and legal arguments, both in written form and through presentations, to academic and professional audiences.
- To instill an understanding of ethical issues in research, including responsible conduct, data integrity, and the ethical use of intellectual property.
- To provide a comprehensive understanding of intellectual property rights, including patents, trademarks, copyrights, and their application in various industries.

UNIT - I	RESEARCH DESIGN	(9)

Overview of research process and design – Use of Secondary and exploratory data to answer the research question, Qualitative research, Observation studies – Experiments and surveys.

UNIT - II	DATA COLLECTION AND SOURCES	(9)

Measurements: Measurement Scales – Questionnaires and Instruments – Sampling and Methods. Data - Preparing, Exploring, Examining and displaying.

UNIT - III	DATA ANALYSIS AND REPORTING	(9)

Overview of Multivariate analysis – Hypotheses testing and Measures of Association – Presenting Insights and findings using written reports and oral presentation.

UNIT - IV	INTELLECTUAL PROPERTY RIGHTS	(9)

Intellectual Property – The concept of IPR, Evolution and development of the concept of IPR, IPR development process, Trade secrets, utility Models, IPR & Biodiversity, Role of WIPO and WTO in IPR establishments, Right of Property, Common rules of IPR practices, Types and Features of IPR Agreement, Trademark, Functions of UNESCO in IPR maintenance.

UNIT - V	PATENTS	(9)

Patents – objectives and benefits of patent – Concept, features of patent, Inventive step, Specification – Types of patent application, process E-filling – Examination of patent – Grant of patent, Revocation, Equitable Assignments. Licenses – Licensing of related patents – patent agents, – Registration of patent agents.

TOTAL: 45 PERIODS

COURSE OUTCOMES:

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

COs	Course Outcome	Cognitive Level
CO1:	Develop a suitable research process to solve real-time problems.	Apply
CO2:	Apply appropriate methods to collect qualitative and quantitative data for analysis.	Apply
CO3:	Apply appropriate statistical tools to analyze data and solve research problems.	Apply
CO4:	Describe the types and features of intellectual property and its role in IPR establishment.	Apply
CO5:	Illustrate the patent procedures, E-filling, register of patents, and licensing of patents.	Apply

TEXT BOOKS:

1	Cooper Donald, R., Schindler Pamela, S., and Sharma, J.K., "Business Research Methods", Tata McGraw Hill Education, Eleventh Edition, 2012.
2	Catherine J. Holland, Intellectual property: Patents, Trademarks, Copyrights, Trade Secrets, Entrepreneur Press, 2007.

REFERENCES:

1	David Hunt, Long Nguyen, Matthew Rodgers, Patent Searching: Tools & Techniques, Wiley, 2007.
2	The Institute of Company Secretaries of India, Statutory body under an Act of Parliament, Professional Programme Intellectual Property Rights, Law and Practice, September 2013.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	3	-	1	1
CO2	3	3	-	1	1
CO3	3	3	-	1	1
CO4	3	3	-	1	1
CO5	3	3	-	1	1
Avg.	3	3	-	1	1

1 - Low, 2 - Medium, 3 - High

BD24T26	BIG DATA MINING AND ANALYTICS	Category	L	T	P	C
		PCC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Basic knowledge of data processing, statistical modeling, and machine learning is required. Familiarity with clustering algorithms and Hadoop is also helpful. Understanding data stream processing and analytics tools will aid in comprehension.

OBJECTIVES

1. To explore statistical modeling, machine learning, and MapReduce for large-scale data.
2. To learn data stream processing techniques such as sampling and filtering.
3. To study and apply various clustering techniques like partitioning and hierarchical methods.
4. To gain practical knowledge of Hadoop's file system and ecosystem for big data management.
5. To master big data analytics and integrate unstructured with structured data.

UNIT – I	LARGE SCALE FILES AND MAPREDUCE	(9)

Statistical Modeling – Machine Learning – Computational Approaches to Modeling – Summarization – Feature Extraction – Statistical Limits on Data Mining – Distributed File Systems – MapReduce – Algorithms Using MapReduce – Extensions to MapReduce.

UNIT – II	MINING DATA STREAMS	(9)

Stream Data Model – Sampling Data in a Stream – Filtering Streams – Counting Distinct Elements in a Stream – Estimating Moments – Counting Ones in a Window – Decaying Windows.

UNIT – III	CLUSTER ANALYSIS AND METHODS	(9)

Cluster Analysis – Partitioning Methods – Hierarchical Methods – Density-Based Methods – Grid-Based Methods – Evaluation of Clustering.

UNIT – IV	HADOOP AND ITS ECOSYSTEM	(9)

Explaining Hadoop – Hadoop Distributed File System – Hadoop MapReduce – Building Hadoop Ecosystem – Managing Resources and Applications with Hadoop YARN – Storing Big Data with HBase – Mining Big Data with Hive.

UNIT – V	ANALYTICS AND BIG DATA	(9)

Defining Big Data Analytics – Exploring Unstructured data – Understanding Text Analytics – Analysis and Extraction Techniques – Putting results together with structured data – Putting Big data to use – Text Analytics tools for Big Data.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Apply statistical modeling, machine learning and MapReduce for large-scale data processing.	**Apply**
CO2	Evaluate stream processing techniques and assess their real-time effectiveness.	**Understand**
CO3	Compare various clustering methods and develop effective models using partitioning, hierarchical, and density-based techniques.	**Analyze**
CO4	Utilize Hadoop's file system and tools, and manage big data with Hadoop YARN.	**Apply**
CO5	Analyze and integrate unstructured and structured data, and apply text analytics tools for insights.	**Analyze**

REFERENCES:

1. Jure Leskovec, Anand Rajaraman, Jeffrey David Ullman, "Mining of Massive Datasets", Cambridge University Press, UK, 3rd Edition, 2020.
2. Jiawei Han, Micheline Kamber, Jian Pei, "Data Mining Concepts and Techniques", Morgan Kaufman Publications, USA, 3rd Edition, 2012.
3. Judith Hurwitz, Alan Nugent, Dr. Fern Halper, and Marcia Kaufman, "Big Data for Dummies", John Wiley and Sons, Inc. New Jersey, 2013.
4. Ian H.Witten, Eibe Frank, "Data Mining – Practical Machine Learning Tools and Techniques", Morgan Kaufman Publications, San Francisco, 3rd Edition, 2011.
5. Seema Acharya, Subhashini Chellappan, "Big Data and analytics", Wiley Publications, India, 1st edition, 2015.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	2	-	3	3	2
CO2	2	-	3	3	2
CO3	2	-	3	3	2
CO4	2	-	3	3	2
CO5	2	-	3	3	2
Avg.	**2**	**-**	**3**	**3**	**2**

1-low, 2-medium, 3-high

CS24T27	CLOUD COMPUTING	Category	L	T	P	C
		PCC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

A basic understanding of cloud computing concepts and deployment models is needed. Familiarity with security principles and privacy concerns in IT is also required. Knowledge of major cloud platforms and their applications will be beneficial.

OBJECTIVES:

1. To Know the cloud computing concepts, deployment models, and architectural design.
2. To discover cloud deployment models and address security issues.
3. To examine privacy concerns and regulatory implications in cloud computing.
4. To learn about major industrial cloud platforms and their applications
5. To study cloud security measures, including infrastructure protection and privacy issues.

UNIT – I	CLOUD PLATFORM ARCHITECTURE	(9)

Cloud Computing: Definition, Characteristics – Cloud deployment models: public, private, hybrid, community – Categories of cloud computing – Everything as a service: Infrastructure, platform, software – A Generic Cloud Architecture Design – Layered cloud Architectural Development – Architectural Design Challenges.

UNIT – II	CLOUD DEPLOYMENT MODELS AND SECURITY ISSUES	(9)

Key Drivers to Adopting the Cloud – The Impact of Cloud Computing on Users – Governance in the Cloud – Barriers to Cloud Computing Adoption in the Enterprise. Infrastructure Security: Network Level – Host Level – Application Level – Data Security and Storage – Aspects of Data Security – Data Security Mitigation Provider Data and Security.

UNIT – III	PRIVACY ISSUES	(9)

Privacy Issues – Data Life Cycle – Key Privacy Concerns in the Cloud – Protecting Privacy – Changes to Privacy Risk Management and Compliance in Relation to Cloud Computing – Legal and Regulatory Implications – U.S. Laws and Regulations – International Laws and Regulations.

UNIT – IV	INDUSTRIAL PLATFORMS AND APPLICATIONS	(9)

Amazon web services – Google App Engine – Microsoft Azure – Scientific applications – Business and consumer applications.

UNIT – V	CLOUD SECURITY	(9)

Cloud Infrastructure security: network, host and application level – Aspects of data security, provider data and its security, Identity and access management architecture, IAM practices in the cloud, SaaS, PaaS, IaaS availability in the cloud – Key privacy issues in the cloud – Cloud Security and Trust Management

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Recognize cloud architecture and design solutions	**Understand**
CO2	Examine cloud models and evaluate security issues.	**Understand**
CO3	Identify privacy concerns and apply regulations.	**Understand**
CO4	Compare cloud platforms and implement applications.	**Apply**
CO5	Make use of cloud security and implement protection measures.	**Apply**

REFERENCES:

1. Rajkumar Buyya, Christian Vacchiola, S.Thamarai Selvi, "Mastering Cloud Computing", MCGraw Hill Education, India, 1st Edition, 2013.
2. John W.Ritting house and James F.Ransome, "Cloud Computing: Implementation, Management, and Security", CRC Press, Florida, 2010.
3. Tim Mather, "Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance (Theory in Practice)", O'Reilly Media, California, 2nd Edition, 2009.
4. John Rhoton, "Cloud Computing Explained: Implementation Handbook for Enterprises", Saint Louis, New York, 1st, 2009.
5. Tom White, "Hadoop: The Definitive Guide", Yahoo Press, New York, 3rd Edition, 2012.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	-	2	3	2
CO2	3	-	2	3	2
CO3	3	-	2	3	2
CO4	3	-	2	3	2
CO5	3	-	2	3	2
Avg.	**3**	**-**	**2**	**3**	**2**

BD24E08	INTERNET OF THINGS	Category	L	T	P	C
		PCC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Basic knowledge of computer networking, familiarity with Python programming, and experience with microcontroller platforms like Arduino or Raspberry Pi. Understanding of web technologies and cloud computing fundamentals.

OBJECTIVES:

1. To introduce the basics of IoT and its underlying technologies.
2. To review the development of IoT systems using popular hardware and software tools.
3. To explore various IoT communication protocols.
4. To study cloud services for IoT applications.
5. To learn Industrial IoT (IIoT) and its applications.

UNIT – I	INTRODUCTION TO INTERNET OF THINGS	(9)

Basic computer networking to Internet of things: Network Types – Layered network models – Addressing – TCP/IP transport Layer. Definition and Characteristics of IoT – Physical Design of IoT – Logical Design of IoT – IoT Enabling Technologies – IoT Levels and Deployment Templates – IoT and M2M.

UNIT – II	BUILDING IOT SYSTEMS	(9)

IoT Physical devices and Endpoints: Basic building blocks of IoT Device – Raspberry Pi – Linux on Raspberry Pi – Interfaces – Programming Raspberry Pi with Python – Python packages for IOT: JSON – XML – HTTPLib – URLLib – SMTPLib – XMPP – Contiki OS – Other IoT Platform: Arduino – Intel Galileo and Beaglebone boards.

UNIT – III	IOT PROTOCOLS	(9)

Introduction to IoT Protocols – 6LoWPAN – IEEE 802.11 – WiFi – 802.15 Bluetooth – 802.15.4 – Zigbee – CoAP.

UNIT – IV	CLOUD OFFERINGS AND IOT CASE STUDIES	(9)

Cloud Storage Models and Communication APIs for IoT– WAMP – Xively Cloud – Python Web Application framework – Designing a RESTful Web API – Amazon Web Services for IoT – MQTT – Case studies for IoT Design: Home automation – Smart Agriculture.

UNIT – V	INDUSTRIAL INTERNET OF THINGS (IIOT)	(9)

Introduction – Industrial Process – The Computer Integrated Manufacturing Pyramid (CIM) – IIoT data flow – Understanding the IIoT edge: Features of the edge – Architecture and implementations. Implementing IOT industrial solution with cloud services.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Comprehend the fundamental concepts of IoT.	**Understand**
CO2	Build simple IoT systems using devices like Raspberry Pi.	**Apply**
CO3	Familiarize with key IoT communication protocols.	**Understand**
CO4	Develop cloud-integrated IoT applications.	**Apply**
CO5	Apply IIoT concepts in industrial scenarios.	**Apply**

REFERENCES:

1. Arshdeep Bahga, Vijay Madisetti, “Internet of Things: A hands-on Approach”, University Press, UK, 1st Edition, 2015.
2. Giacomo Veneri, Antonio Capasso, “Hands-On Industrial Internet of Things Create a Powerful Industrial IoT Infrastructure Using Industry 4.0”, Packt Publishing Ltd, UK, 1st Edition, 2018.
3. Adrian McEwen Hakim Cassimally, “Designing the Internet of Things", Wiley, India, 1st Edition, 2013.
4. Olivier Hersent, David Boswarthick, Omar Elloum, “The Internet of Things – Key applications and Protocols”, Wiley, India, 1st Edition, 2012.
5. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stamatis, Karnouskos, Stefan Avesand, David Boyle, “From Machine - to-Machine to the Internet of Things - Introduction to a New Age of Intelligence”, Academic Press, Elsevier Science, 1st Edition, 2014.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	2	1	2	3	2
CO2	2	1	2	3	2
CO3	2	1	2	3	2
CO4	2	1	2	3	2
CO5	2	1	2	3	2
Avg.	**2**	**1**	**2**	**3**	**2**

BD24P26	BIG DATA MINING AND ANALYTICS LABORATORY	Category	L	T	P	C
		PCC	0	0	3	2

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Students should have a basic understanding of big data concepts, programming skills (Java or Python), familiarity with data analysis tools (R) and knowledge of NoSQL databases (HBase, MongoDB). Additionally, they should be acquainted with distributed computing frameworks, particularly Apache Spark.

OBJECTIVES:

1. To install, configure, and run Hadoop and HDFS for big data processing.
2. To develop and implement MapReduce programs for data processing tasks such as word count and weather dataset analysis.
3. To apply SVM and clustering techniques using R for data analysis and visualization.
4. To implement applications that store and manage big data in HBase or MongoDB using Hadoop or R.
5. To install, deploy, and configure an Apache Spark cluster and execute applications using Spark.

List of Exercise/Experiments:

1. Install, configure and run Hadoop and HDFS.
2. Develop and execute MapReduce programs to count word frequencies.
3. Create a MapReduce program to process weather data.
4. Implement SVM and clustering techniques using R.
5. Visualize data using any plotting framework.
6. Build an application that stores big data in HBase or MongoDB using Hadoop or R.
7. Set up and configure an Apache Spark cluster, and run an application using Apache Spark

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Demonstrate the ability to install, configure, and effectively manage Hadoop and HDFS for big data environments.	Apply

CO2	Develop and execute MapReduce programs for efficient data processing.	Apply
CO3	Apply SVM and clustering techniques in R and visualize data effectively.	Apply
CO4	Implement and manage storage solutions for large datasets using HBase or MongoDB, integrating them with Hadoop or R.	Apply
CO5	Deploy and use Apache Spark for distributed data processing applications.	Apply

REFERENCES:

1. Jure Leskovec, Anand Rajaraman, Jeffrey David Ullman, "Mining of Massive Datasets", Cambridge University Press, UK, 3rd Edition, 2020.
2. Seema Acharya, Subhashini Chellappan, "Big Data and analytics", Wiley Publications, India, 1st edition, 2015.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	2	-	3	3	2
CO2	2	-	3	3	2
CO3	2	-	3	3	2
CO4	2	-	3	3	2
CO5	2	-	3	3	2
Avg.	**2**	**-**	**3**	**3**	**2**

1-low, 2-medium, 3-high

CS24P26	CLOUD COMPUTING LABORATORY	Category	L	T	P	C
		PCC	0	0	3	2

(Common to M.E CSE and M.E BDA)

PREREQUISITE

A basic understanding of virtualization, network protocols, and client-server architecture is required. Familiarity with network design tools and cloud computing concepts is also necessary.

OBJECTIVES:

1. To install and configure hypervisors; instantiate VMs.
2. To facilitate client-server communication between virtual machines by developing and executing a chat application.
3. To design and implement simple network topologies using network virtualization tools.
4. To provide students with hands-on experience in implementing network protocols using network controllers.
5. To develop students' ability to analyze and optimize scheduling mechanisms in cloud environments using simulation tools.

List of Exercise/Experiments:

1. a) Installation of various hypervisors and instantiation of VMs with image file using open source hypervisors such as Virtual Box, VMWare Player, Xen and KVM.
   b) Client server communication between two virtual machine instances, execution of chat application.
2. Creation of simple network topology using open source network virtualization tools (like mini net and others).
3. Implementation of simple network protocols using open source network controllers (like Open Daylight).
4. Implementation of various scheduling mechanisms using open source cloud simulator.
5. Familiarization and usage of the following cloud services with open source cloud tools (like Eucalyptus, Open stack, Open Nebula and others)
   a. Scheduling mechanisms
   b. Load balancing mechanisms
   c. Hashing and encryption mechanisms
6. Familiarization and usage of collaborative applications (SaaS).
7. Implementing applications using Google App Engine (PaaS).
   a. Develop MapReduce application (example-URL Pattern count and others) using Hadoop cluster set up (Single node and multi node).

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Apply hypervisor installation and VM creation techniques.	Apply
CO2	Construct and manage VM client-server communication.	Apply
CO3	Design and implement basic network topologies.	Apply
CO4	Implement and manage network protocols with controllers.	Apply
CO5	Examine scheduling mechanisms in cloud environments to optimize resource allocation using simulation tools.	Apply

REFERENCES:

1. Rajkumar Buyya, Christian Vacchiola, S.Thamarai Selvi, "Mastering Cloud Computing" , MCGraw Hill Education, India, 1st Edition, 2013.
2. John W.Rittinghouse and James F.Ransome, "Cloud Computing: Implementation, Management, and Security", CRC Press, Florida, 2010.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	1	3	3	2
CO2	3	1	3	3	2
CO3	3	1	3	3	2
CO4	3	1	3	3	2
CO5	3	1	3	3	2
Avg.	**3**	**1**	**3**	**3**	**2**

1-low, 2-medium, 3-high

CS24P21	TECHNICAL PRESENTATION	Category	L	T	P	C
		EEC	0	0	3	2

PREREQUISITE

Students should start by conducting thorough research on their chosen topic, reviewing recent journals and conference papers. They must select their topic with guidance from faculty to ensure relevance. Additionally, students need to develop strong presentation skills to clearly and effectively communicate their findings, using appropriate visual aids.

OBJECTIVES:

1. To show expertise in the chosen topic with recent research.
2. To provide original insights based on critical analysis.
3. To communicate complex ideas clearly to the audience.
4. To align the topic with relevant program outcomes.
5. To encourage feedback and discussion to refine ideas.

Guidelines:

1. The students have to refer the journals, conference proceedings which are published recently.
2. By mutual discussions with the faculty, the student can choose a topic in specific area.
3. The student has to submit a technical report having 30 - 50 pages to the corresponding faculty one week before the final presentation.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Deepen understanding of the topic and its relevance to recent research.	Understand
CO2	Present technical information clearly and confidently.	Understand
CO3	Conduct independent research and produce a coherent report.	Understand
CO4	Assess current research and identify future opportunities.	Understand
CO5	Receive feedback to refine research and presentation skills.	Understand

Mapping of COs with POs and PSOs					
COs/ POs	**PO1**	**PO2**	**PO3**	**PSO1**	**PSO2**
CO1	**3**	**1**	**3**	**3**	**2**
CO2	**3**	**1**	**3**	**3**	**2**
CO3	**3**	**1**	**3**	**3**	**2**
CO4	**3**	**1**	**3**	**3**	**2**
CO5	**3**	**1**	**3**	**3**	**2**
Avg.	**3**	**1**	**3**	**3**	**2**
1-low, 2-medium, 3-high					

CS24E01	DATA MINING TECHNIQUES (PROFESSIONAL ELECTIVES – I and II)	Category	L	T	P	C
		PEC	3	0	0	3

PREREQUISITE

Basic knowledge of statistics, programming, and databases is needed. Data warehousing requires understanding of databases and data modeling. Classification methods need basics of statistics, databases, and machine learning. Clustering and association rule mining require statistics, machine learning and programming skills.

OBJECTIVES:

1. To present data mining concepts and preprocessing techniques.
2. To study data warehouse fundamentals and design.
3. To investigate various classification methods, including decision trees, Bayesian methods and support vector machines.
4. To study different clustering methods and techniques.
5. To learn about association rule mining and data visualization

UNIT – I	DATA MINING AND DATA PREPROCESSING	(9)

Introduction to Data Mining – Kinds of Data – Kinds of Patterns – Technologies – Kinds of Applications – Major Issues in Data Mining – Data Preprocessing: An Overview – Data Cleaning – Data Integration – Data Reduction – Data Transformation and Data Discretization.

UNIT – II	BASICS OF DATA WAREHOUSE	(9)

Basic Concepts – Data Warehouse Modeling – Data Warehouse Design and Usage – Data Warehouse Implementation – Data Generalization by Attribute Oriented Induction.

UNIT – III	CLASSIFICATIONS	(9)

Classifications – Basic Concepts – Decision Tree induction – Bayes Classification Methods – Rule Based Classification – Model Evaluation and Selection – Techniques to Improve Classification Accuracy – Classification: Advanced concepts – Bayesian Belief Networks – Classification by Back Propagation – Support Vector Machine – Classification using frequent patterns.

UNIT – IV	CLUSTER ANALYSIS	(9)

Cluster Analysis: Basic concepts and Methods – Cluster Analysis – Partitioning methods – Hierarchical methods – Density Based Methods – Grid Based Methods – Evaluation of Clustering – Advanced Cluster Analysis: Probabilistic model-based clustering – Clustering High – Dimensional Data – Clustering Graph and Network Data – Clustering with Constraints.

UNIT – V	ASSOCIATION RULE MINING AND VISUALIZATION	(9)

Basic Concepts – Frequent Itemset Mining Methods – Pattern Evaluation Methods – Pattern Mining – Pattern Mining in Multilevel, Multidimensional Space – Constraint-Based Frequent Pattern Mining – C Mining High-Dimensional Data and Colossal Patterns – Mining Compressed or Approximate Patterns – Pattern Exploration and Application – Data Visualization – Case Study: WEKA.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Comprehend data mining methods and apply preprocessing techniques.	**Understand**
CO2	Recognize data warehousing principles and implement design practices.	**Understand**
CO3	Identify classification algorithms, apply them to datasets, and assess their performance.	**Apply**
CO4	Discover clustering methods and evaluate their effectiveness.	**Apply**
CO5	Realize mining and visualization techniques and review their effectiveness.	**Understand**

REFERENCES:

1. Jaiwei Han, Micheline Kamber and Jian Pei, "Data Mining Concepts and Techniques", Morgan Kauffman, 3 rd Edition, 2012.
2. K.P. Soman, Shyam Diwakar and V. Ajay, "Insight into Data mining Theory and Practice", PHI/Eastern Economy, 5th Edition, 2014.
3. Alex Berson and Stephen J.Smith, "Data Warehousing, Data Mining & OLAP", Tata McGraw – Hill, 5th Edition, 2016.
4. G. K. Gupta, "Introduction to Data Mining with Case Studies", Prentice Hall of India, 3 rd Edition, 2014.
5. Ian H.Witten and Eibe Frank, Mark A. Hall, "Data Mining: Practical Machine Learning Tools and Techniques", Elsevier, 3 rd Edition, 2011.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	-	2	3	1
CO2	3	-	2	3	1
CO3	3	-	2	3	1
CO4	3	-	2	3	1
CO5	3	-	2	3	1
Avg.	**3**	**-**	**2**	**3**	**1**

1-low, 2-medium, 3-high

CS24E02	ADVANCED OPERATING SYSTEM (PROFESSIONAL ELECTIVES – I and II)	Category	L	T	P	C
		PEC	3	0	0	3

PREREQUISITE

Students should have a solid understanding of basic operating system concepts, proficiency in programming languages like C or C++, and familiarity with computer architecture. Experience with data structures, algorithms and system-level programming is also essential.

OBJECTIVES:

1. To gain insight into the design and management of multiprocessor systems.
2. To explore the frameworks and challenges of distributed operating systems.
3. To investigate the strategies for managing resources in distributed systems.
4. To study database OS models and focus on transaction processing and concurrency control.
5. To learn about real-time systems and mobile OS architectures.

UNIT – I	MULTIPROCESSOR SYSTEM ARCHITECTURES	(9)

Multiprocessor Operating Systems: Motivation for multiprocessor Systems – Multiprocessor System Architectures – Operating system Design Issues – Threads – Process Synchronization – Processor Scheduling and Allocation – memory management.

UNIT – II	DISTRIBUTED OPERATING SYSTEMS	(9)

Architectures of Distributed Systems: System Architecture – Issues in Distributed Operating Systems – Communication Primitives. Theoretical Foundations: Inherent Limitations of a Distributed System – Lamport's Logical Clocks – Vector Clocks – Causal Ordering of Messages – Distributed Deadlock Detection: Centralized, Distributed and Hierarchical Deadlock Detection Algorithms.

UNIT – III	DISTRIBUTED RESOURCE MANAGEMENT	(9)

Distributed File Systems: Architecture – Mechanisms for Building Distributed File Systems – Design Issues. Distributed Shared Memory: Architecture and Motivation – Algorithms for Implementing DSM – Memory Coherence – Coherence Protocols. Distributed Scheduling: Issues in Load Distributing – Components of a Load Distributed Algorithm – Stability – Load Distributing Algorithms – Requirements for Load Distributing – Task Migration – Issues in task Migration.

UNIT – IV	DATABASE OPERATING SYSTEMS	(9)

Database Operating Systems: Requirements of Database OS – Transaction process model – Synchronization primitives – Concurrency control algorithms.

UNIT – V	MOBILE AND REAL TIME OPERATING SYSTEMS	(9)

Basic Model of Real Time Systems – Characteristics – Applications of Real Time Systems – Real Time Task Scheduling – Handling Resource Sharing – Mobile Operating Systems – Architecture – Layers – Microkernel Design – Kernel Extensions – Processes and Threads – Memory Management – File system – Android – iOS.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Describe the principles and administration of multiprocessor systems.	**Understand**
CO2	Examine distributed OS frameworks and their challenges.	**Understand**
CO3	Develop and evaluate resource management strategies for distributed systems.	**Understand**
CO4	Outline database OS models and implement transaction processing and concurrency control.	**Understand**
CO5	Identify and demonstrate real-time and mobile OS architectures.	**Apply**

REFERENCES:

1. Mukesh Singhal, Niranjan Shivaratri, "Advanced Concepts in Operating Systems – Distributed, Database and Multiprocessor Operating Systems", Tata McGraw-Hill, 1st Edition, 2011.
2. Andrew S. Tanenbaum and Herbert Bos, "Modern Operating Systems", Prentice Hall, 4th Edition, 2014
3. A S Tanenbaum, "Distributed Operating Systems", Pearson Education, India, 5th Edition, 2008.
4. Rajib Mall, "Real-Time Systems: Theory and Practice", Prentice Hall, 2nd Edition, 2006.
5. Neil Smyth, "iPhone iOS 4 Development Essentials – Xcode", Payload Media, 4th Edition, 2011.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	2	3	3	2
CO2	3	2	3	3	2
CO3	3	2	3	3	2
CO4	3	2	3	3	2
CO5	3	2	3	3	2
Avg.	**3**	**2**	**3**	**3**	**2**

1-low, 2-medium, 3-high

CS24E03	MOBILE AND PERVASIVE COMPUTING (PROFESSIONAL ELECTIVES – I and II)	Category	L	T	P	C
		PEC	3	0	0	3

PREREQUISITE

It requires programming skills, knowledge of networking and operating systems, and familiarity with mobile app development. Understanding distributed systems is also important for grasping pervasive computing concepts.

OBJECTIVES:

1. To study wireless communication and the evolution from 2G to 5G.
2. To explore mobile communication technologies and network support.
3. To learn pervasive communication and application architectures.
4. To assess the principles and implementations of context-aware computing systems.
5. To review context-aware sensor networks and future innovations.

UNIT – I	INTRODUCTION TO WIRELESS ENVIRONMENT	(9)

Introduction to wireless communication – Wireless Transmission – Medium Access Control – Wireless MAC protocols – Comparison of 2G, 3G, 4G looking ahead 5G systems.

UNIT – II	MOBILE COMMUNICATION	(9)

GSM – Bluetooth – Mobile network layer – Mobile transport layer – File system support for mobility support – Mobile execution environments and applications.

UNIT – III	PERVASIVE COMMUNICATION	(9)

Past, Present, Future – Application Examples – Device Technology – WAP and Beyond – Pervasive Web Application Architecture: Example Application.

UNIT – IV	CONTEXT AWARE COMPUTING	(9)

Structure and Elements of Context-aware Pervasive Systems: Abstract architecture – Infrastructures – Middleware and toolkits. Context-aware mobile services: Context for mobile device users – Location-based services – Ambient service – Enhancing Context – Aware mobile services and Context aware artifacts.

UNIT – V	CONTEXT AWARE PERVASIVE SYSTEM	(9)

Context-aware sensor networks – A framework for Context aware sensors – Context-aware security systems – Constructing Context-aware pervasive system – Future of Content aware systems.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Summarize wireless communication and compare wireless generations.	**Understand**
CO2	Describe mobile technologies and evaluate network support.	**Understand**
CO3	Explore pervasive communication and design web applications.	**Apply**
CO4	Analyze context-aware systems and implement mobile services.	**Apply**
CO5	Investigate sensor networks and predict future trends.	**Understand**

REFERENCES:

1. Schiller Jochen, "Mobile Communication", PHI/Pearson Education, India, 2nd Edition, 2009.
2. Burkhardt Jochen, Henn Horst and Hepper Stefan, Schaec Thomas and Rindtorff Klaus, "Pervasive Computing Technology and Architecture of Mobile Internet Applications", Addison Wesley Reading, India, 2007.
3. Seng Loke, "Context-Aware Pervasive Systems: Architectures for a New Breed of Applications", Auerbach Publications, New York, 1st Edition, 2006.
4. Natalia Silvis, "Pervasive Computing Engineering Smart Systems", Springer, Netherland, 1st Edition, 2017.
5. Frank Adelstein, "Fundamentals of Mobile and Pervasive Computing", TMH, India, 1st Edition, 2005.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	-	3	3	1
CO2	3	-	3	3	1
CO3	3	-	3	3	1
CO4	3	-	3	3	1
CO5	3	-	3	3	1
Avg.	**3**	**-**	**3**	**3**	**1**

1-low, 2-medium, 3-high

BD24T16	FOUNDATIONS OF DATA SCIENCE (PROFESSIONAL ELECTIVES – I and II)	Category	L	T	P	C
		PEC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

To succeed in a Foundations of Data Science course, students should have basic programming skills in languages like Python or R, a foundational understanding of statistics and mathematics (including linear algebra and calculus), and familiarity with data manipulation and databases.

OBJECTIVES:

1. To introduce data science fundamentals and processes.
2. To study methods for statistical description and analysis of data.
3. To explore correlation and regression techniques.
4. To learn data manipulation using Python libraries.
5. To develop skills in creating and customizing data visualizations using Python tools.

UNIT – I	BASICS OF DATA SCIENCE	(9)

Data Science: Benefits and uses – Facets of data – Data Science Process: Overview – Defining research goals – Retrieving data – Data preparation – Exploratory Data analysis – Build the model – Presenting findings and building applications – Data Mining – Data Warehousing – Basic Statistical descriptions of Data.

UNIT – II	DESCRIBING DATA	(9)

Types of Data – Types of Variables – Describing Data with Tables and Graphs – Describing Data with Averages Variability – Normal Distributions and Standard (z) Scores.

UNIT– III	DESCRIBING RELATIONSHIP	(9)

Correlation – Scatter plots – Correlation coefficient for quantitative data – Computational formula for correlation coefficient – Regression – Regression line – Least squares regression line – Standard error of estimate – Interpretation of r2 – Multiple regression equations – Regression towards the mean.

UNIT – IV	PYTHON LIBRARIES FOR DATA WRANGLING	(9)

Basics of Numpy arrays – Aggregations – Computations on arrays – Comparisons, Masks, Boolean logic – Fancy indexing – Structured arrays – Data manipulation with Pandas – Data indexing and selection – Operating on data – Missing data – Hierarchical indexing – Combining datasets – Aggregation and Grouping – Pivot tables.

UNIT – V	DATA VISUALIZATION	(9)

Importing Matplotlib – Line plots – Scatter plots – Visualizing errors – Density and contour plots – Histograms – Legends – Colors – Subplots – Text and Annotation – Customization – Three-Dimensional plotting – Geographic Data with Basemap – Visualization with Seaborn.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Enlighten data science fundamentals and processes.	**Understand**
CO2	Summarize data using statistical methods.	**Understand**
CO3	Apply and interpret correlation3 and regression techniques.	**Apply**
CO4	Manipulate data using Python libraries.	**Apply**
CO5	Analyze and customize data visualizations in Python.	**Analyze**

REFERENCES:

1. David Cielen, Arno D. B. Meysman, and Mohamed Ali, "Introducing Data Science", Manning Publications, New Yark, 1st Edition, 2016.
2. Robert S. Witte and John S. Witte, "Statistics", Wiley Publications, India, 11th Edition, 2021.
3. Jake VanderPlas, "Python Data Science Handbook", O'Reilly, US, 1st Edition, 2016.
4. Allen B. Downey, "Think Stats: Exploratory Data Analysis in Python", Green Tea Press, India, 2nd Edition, 2016.
5. Sinan Ozdemir, "Principles of Data Science", Packt Publication, UK, 3rd Edition, 2024.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	1	1	3	3	2
CO2	1	1	3	3	2
CO3	1	1	3	3	2
CO4	1	1	3	3	2
CO5	1	1	3	3	2
Avg.	**1**	**1**	**3**	**3**	**2**

1-low, 2-medium, 3-high

BD24E04	AGILE METHODOLOGIES (PROFESSIONAL ELECTIVES – I and II)	Category	L	T	P	C
		PEC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Success in this course requires a basic understanding of software development and software engineering concepts, including requirements and testing. Familiarity with traditional project management is helpful for contrasting with Agile. Experience in team collaboration and an interest in iterative development are also important.

OBJECTIVES:

1. To provide the core theories and principles of Agile methodologies.
2. To learn about various Agile processes like SCRUM and Extreme Programming.
3. To develop skills to manage and share knowledge effectively within Agile teams.
4. To gain expertise in handling requirements engineering in an Agile context.
5. To learn Agile approaches to quality assurance, including Test Driven Development.

UNIT – I	BASICS OF AGILE METHODOLOGY	(9)

Theories for Agile Management – Agile Software Development – Traditional Model vs. Agile Model – Classification of Agile Methods – Agile Manifesto and Principles – Agile Project Management – Agile Team Interactions – Ethics in Agile Teams – Agility in Design, Testing – Agile Documentations – Agile Drivers, Capabilities and Values.

UNIT – II	AGILE PROCESSES	(9)

Lean Production – SCRUM, Crystal, Feature Driven Development – Adaptive Software Development – Extreme Programming: Method Overview – Lifecycle – Work Products, Roles and Practices.

UNIT– III	AGILITY AND KNOWLEDGE MANAGEMENT	(9)

Agile Information Systems – Agile Decision Making – Earl's Schools of KM – Institutional Knowledge Evolution Cycle – Development, Acquisition, Refinement, Distribution, Deployment, leveraging – KM in Software Engineering – Managing Software Knowledge – Challenges of Migrating to Agile Methodologies – Agile Knowledge Sharing – Role of Story-Cards – Story-Card Maturity Model (SMM).

UNIT – IV	AGILITY AND REQUIREMENTS ENGINEERING	(9)

Impact of Agile Processes in RE – Current Agile Practices – Variance – Overview of RE Using Agile – Managing Unstable Requirements – Requirements Elicitation – Agile Requirements Abstraction Model – Requirements Management in Agile Environment – Agile Requirements Prioritization – Agile Requirements Modeling and Generation – Concurrency in Agile Requirements Generation.

UNIT – V	AGILITY AND QUALITY ASSURANCE	(9)

Agile Product Development – Agile Metrics – Feature Driven Development (FDD) – Financial and Production Metrics in FDD – Agile Approach to Quality Assurance – Test Driven Development – Agile Approach in Global Software Development.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Apply Agile methodologies in real-world projects.	**Apply**
CO2	Use Agile tools like SCRUM and FDD effectively.	**Understand**
CO3	Effectively manage and distribute knowledge in Agile environments.	**Understand**
CO4	Handle Agile requirements engineering with proficiency.	**Understand**
CO5	Implement Agile quality assurance to ensure high-quality software.	**Apply**

REFERENCES:

1. Dingsoyr, Torgeir, Dyba, Tore, Moe, Nils Brede (Eds.), "Agile Software Development, Current Research and Future Directions", Springer-Verlag Berlin Heidelberg, UK, 1st Edition, 2010.
2. David J. Anderson; Eli Schragenheim, "Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results", Prentice Hall, India, 1st Edition, 2003
3. Hazza & Dubinsky, "Agile Software Engineering, Series: Undergraduate Topics in Computer Science", Springer, UK, 8th Edition, 2009
4. Craig Larman, "Agile and Iterative Development: A managers Guide", Addison-Wesley, New York, 2nd Edition, 2004
5. Kevin C. Desouza, "Agile information systems: conceptualization, construction, and management, Butterworth-Heinemann", Penguin Books Ltd, UK, 1st Edition, 2007.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	1	2	2	3
CO2	3	1	2	2	3
CO3	3	1	2	2	3
CO4	3	1	2	2	3
CO5	3	1	2	2	3
Avg.	**3**	**1**	**2**	**2**	**3**

1-low, 2-medium, 3-high

CS24E04	OBJECT ORIENTED SOFTWARE ENGINEERING (PROFESSIONAL ELECTIVES – I and II)	Category	L	T	P	C
		PEC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Students should have basic programming skills in an object-oriented language like Java or C++ and a solid understanding of software engineering principles. Familiarity with core data structures and algorithms is essential, along with experience using UML for modeling designs.

OBJECTIVES:

1. To introduce core software engineering concepts and Agile methodologies.
2. To teach skill for gathering, analyzing, and specifying software requirements.
3. To cover software design principles, patterns, and architectural styles.
4. To provide knowledge of testing methods and debugging techniques.
5. To learn skills in software project management and DevOps practices.

UNIT – I	SOFTWARE PROCESS AND AGILE DEVELOPMENT	(9)

Introduction to Software Engineering – Software Process – Perspective and Specialized Process Models – Introduction to Agility – Agile process – Extreme programming – XP Process.

UNIT – II	REQUIREMENTS ANALYSIS AND SPECIFICATION	(9)

Requirement analysis and specification – Requirements gathering and analysis – Software Requirement Specification – Formal system specification – Finite State Machines – Petrinets – Object modelling using UML – Use case Model – Class diagrams – Interaction diagrams – Activity diagrams – State chart diagrams – Functional modelling – Data Flow Diagram – Case Tools.

UNIT – III	SOFTWARE DESIGN	(9)

Software design – Design process – Design concepts – Coupling – Cohesion – Functional independence – Design patterns – Model-view-controller – Publish-subscribe – Adapter – Command – Strategy – Observer – Proxy – Facade – Architectural styles – Layered – Client Server – Tiered Pipe and filter – User interface design.

UNIT – IV	SOFTWARE TESTING AND MAINTENANCE	(9)

Testing – Unit testing – Black box testing – White box testing – Integration and System testing – Regression testing – Debugging – Program analysis – Symbolic execution – Model Checking.

UNIT – V	PROJECT MANAGEMENT	(9)

Software Project Management – Software Configuration Management – Project Scheduling – DevOps: Motivation – Cloud as a platform – Operations – Deployment Pipeline: Overall Architecture Building and Testing – Deployment – Tools – Case Study.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Apply fundamental software engineering concepts and Agile methodologies to projects.	**Apply**
CO2	Develop and analyze software requirements.	**Analyze**
CO3	Design software systems using appropriate design patterns and architectural styles.	**Apply**
CO4	Perform various testing and maintenance activities to ensure software quality.	**Understand**
CO5	Manage software projects, including scheduling, configuration, and applying DevOps practices.	**Apply**

REFERENCES:

1. Bernd Bruegge and Allen H. Dutoit, "Object-Oriented Software Engineering: Using UML, Patterns and Java", Pearson Education, India, 3rd Edition, 2013.
2. Roger S. Pressman, "Object-Oriented Software Engineering: An Agile Unified Methodology", Mc Graw-Hill, 1st Edition, 2014
3. Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, "Fundamentals of Software Engineering", PHI Learning Pvt. Ltd, 2nd Edition, 2010.
4. Len Bass, Ingo Weber and Liming Zhu, "DevOps: A Software Architect's Perspective", Pearson Education, 2016.
5. Stephen Schach, "Object-Oriented and Classical Software Engineering", McGraw-Hill, 8th Edition, 2010

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	1	3	2	3
CO2	3	1	3	2	3
CO3	3	1	3	2	3
CO4	3	1	3	2	3
CO5	3	1	3	2	3
Avg.	**3**	**1**	**3**	**2**	**3**

1-low, 2-medium, 3-high

CS24E05	WIRELESS SENSOR NETWORKS (PROFESSIONAL ELECTIVES – I and II)	Category	L	T	P	C
		PEC	3	0	0	3

PREREQUISITE

Basic understanding of wireless communication principles and networking concepts. Familiarity with programming and design in embedded systems is essential. Prior knowledge of network protocols and security fundamentals is also beneficial.

OBJECTIVES:

1. To learn the fundamentals of wireless sensor network design and communication standards.
2. To study MAC and routing protocols, including their applications and performance.
3. To explore transport protocols and Quality of Service (QoS) in sensor networks, focusing on congestion control, in-network processing, and related operating systems.
4. To examine security issues, including attacks, key management, and security protocols.
5. To learn to use and apply tools and programming environments for developing and simulating sensor networks.

UNIT – I	WIRELESS SENSOR NETWORK ARCHITECTURE	(9)

Introduction to wireless sensor networks – Challenges – Comparison with ad hoc network – Node architecture and Network architecture – Design principles – Service interfaces – Gateway – Short range radio communication standards – Physical layer and transceiver design considerations.

UNIT – II	MAC AND ROUTING IN WIRELESS SENSOR NETWORKS	(9)

Introduction – Applications – Challenges – Sensor network architecture – MAC Protocols for wireless sensor networks – Low duty cycle protocols and wakeup concepts – Contention-Based protocols – Schedule-Based protocols – IEEE 802.15.4 Zig bee – Topology Control – Routing Protocols.

UNIT – III	TRANSPORT AND QOS IN WIRELESS SENSOR NETWORKS	(9)

Data-Centric and Contention-Based Networking – Transport Layer and QoS in Wireless Sensor Networks – Congestion Control – In-network processing – Operating systems for wireless sensor networks.

UNIT – IV	SECURITY IN AD HOC AND SENSOR NETWORKS	(9)

Security Attacks – Key Distribution and Management – Intrusion Detection – Software based Antitamper techniques – Watermarking techniques – Defense against routing attacks – Secure Ad hoc routing protocols – Broadcast authentication WSN protocols – TESLA – Sensor Network Security Protocols – SPINS.

UNIT – V	TOOLS FOR WSN	(9)

TinyOS – Introduction, NesC, Interfaces, modules, configuration, Programming in TinyOS using NesC, TOSSIM, Contiki – Structure, Communication Stack, Simulation environment – Cooja Simulator, Programming.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Describe the fundamentals of wireless sensor network design and communication standards	**Understand**
CO2	Analyze MAC and routing protocols, assessing their applications and performance	**Analyze**
CO3	Apply knowledge of transport protocols and Quality of Service (QoS) mechanisms to manage congestion control and in-network processing	**Apply**
CO4	Assess security challenges, including attacks and key management, to address vulnerabilities	**Understand**
CO5	Apply tools and programming environments to develop and simulate sensor networks effectively	**Apply**

REFERENCES:

1. Anna Hac, "Wireless Sensor Network Design", John Wiley & Sons, 2003.
2. Holger Karl, Andreas Willig, "Protocols and Architectures for Wireless Sensor Networks", John Wiley & Sons, Inc., 2007.
3. Erdal Çayırcı, Chunming Rong, "Security in Wireless Ad Hoc and Sensor Networks", John Wiley and Sons, 2009.
4. C.Siva Ram Murthy and B.S.Manoj, "Ad Hoc Wireless Networks – Architectures and Protocols", Pearson Education, 1 st 2006.
5. Carlos De Morais Cordeiro, Dharma Prakash Agrawal, "Ad Hoc and Sensor Networks: Theory and Application", World Scientific Publishing, 2nd Edition, 2011.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	2	1	3	2	3
CO2	2	1	3	2	3
CO3	2	1	3	2	3
CO4	2	1	3	2	3
CO5	2	1	3	2	3
Avg.	**2**	**1**	**3**	**2**	**3**

1-low, 2-medium, 3-high

CS24E06	MULTI CORE ARCHITECTURES (PROFESSIONAL ELECTIVES – I and II)	Category	L	T	P	C
		PEC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Students should have a basic understanding of computer architecture and digital logic design. Familiarity with programming concepts and performance measurement techniques is also required. Prior knowledge of parallel computing and memory systems is beneficial.

OBJECTIVES:

1. To learn the basics of computer design, performance measurement, and ILP.
2. To study cache performance, virtual memory, and memory hierarchy design.
3. To examine shared memory architectures, cache coherence, and synchronization.
4. To explore homogeneous and heterogeneous multicore designs and their applications.
5. To assess vector architectures, SIMD extensions, and GPU computing.

UNIT – I	FUNDAMENTALS OF COMPUTER DESIGN AND ILP	(9)

Fundamentals of Computer Design – Measuring and Reporting Performance – Instruction Level Parallelism and its Exploitation – Concepts and Challenges – Limitations of ILP – Multithreading – SMT and CMP Architectures – The Multicore era.

UNIT – II	MEMORY HIERARCHY DESIGN	(9)

Introduction – Optimizations of Cache Performance – Memory Technology and Optimizations – Protection: Virtual Memory and Virtual Machines – Design of Memory Hierarchies – Case Studies.

UNIT – III	MULTIPROCESSOR ISSUES	(9)

Symmetric and Distributed Shared Memory Architectures – Cache Coherence Issues – Performance Issues – Synchronization Issues – Models of Memory Consistency – Interconnection Networks – Buses, Crossbar and Multi-Stage Interconnection Networks.

UNIT – IV	MULTICORE ARCHITECTURES	(9)

Homogeneous and Heterogeneous Multicore Architectures – Intel Multicore Architectures – SUN CMP Architecture – IBM Cell Architecture – Introduction to Warehouse – Scale Computers, Cloud Computing – Architectures and Issues – Case Studies.

UNIT – V	VECTOR AND GPU ARCHITECTURES	(9)

Vector Architecture – SIMD Extensions for Multimedia – Graphics Processing Units – Case Studies – GPGPU Computing – Detecting and Enhancing Loop Level Parallelism.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Describe fundamental concepts of computer design, performance metrics, and instruction-level parallelism	**Understand**
CO2	Apply principles to optimize cache performance and design efficient memory hierarchies	**Apply**
CO3	Analyze issues related to cache coherence, synchronization, and memory consistency in multiprocessor systems	**Analyze**
CO4	Compare different multicore architectures and their effectiveness in various computing environments	**Understand**
CO5	Use knowledge of vector and GPU architectures to improve performance in parallel computing tasks	**Understand**

REFERENCES:

1. John L. Hennessey and David A. Patterson, "Computer Architecture – A Quantitative Approach", Morgan Kaufmann, Elsevier, Netherlands, 5th Edition, 2012.
2. Darryl Gove, "Multicore Application Programming: For Windows, Linux, and Oracle Solaris", Pearson, New Delhi, 2nd Edition, 2011.
3. David B. Kirk, Wen-mei W. Hwu, "Programming Massively Parallel Processors", Elsevier Inc, 1st Edition, 2010.
4. Wen–mei W. Hwu, "GPU Computing Gems", Morgan Kaufmann / Elsevier, 2nd Edition, 2011.
5. KaiHwang, "Advanced Computer Architecture", Tata McGraw-Hill Education, 3rd Edition, 2003.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	2	1	3	3	2
CO2	2	1	3	3	2
CO3	2	1	3	3	2
CO4	2	1	3	3	2
CO5	2	1	3	3	2
Avg.	**2**	**1**	**3**	**3**	**2**

1-low, 2-medium, 3-high

CS24E07	HUMAN COMPUTER INTERACTION (PROFESSIONAL ELECTIVES – I and II)	Category	L	T	P	C
		PEC	3	0	0	3

PREREQUISITE

Students should have a foundational understanding of basic design principles and user experience concepts. Familiarity with web development technologies and mobile app design.

OBJECTIVES:

1. To learn the basics of HCI, ergonomics, and user-centered design.
2. To study interaction styles like GUIs and direct manipulation.
3. To examine techniques for usability testing and user experience assessment.
4. To explore task analysis and interaction models in computing.
5. To gain insights into the principles of designing user-friendly and effective web and mobile interfaces.

UNIT – I	FOUNDATIONS OF HCI	(9)

Context of Interaction – Ergonomics – Designing Interactive systems – Understanding Users cognition and cognitive frameworks, User Centered Approaches Usability, Universal Usability, Understanding and conceptualizing interaction, Guidelines, Principles and Theories. Importance of User Interface: Definition – Importance of good design – Benefits of good design.

UNIT – II	INTERACTION STYLES	(9)

GUI: Popularity of graphics – Concept of direct manipulation – Graphical system – Characteristics – Web user – Interface Popularity – Characteristics and Principles of User Interface. Understanding interaction styles – Direct Navigation and Immersive environments – Fluid navigation – Expressive Human and Command Languages.

UNIT – III	EVALUATION OF INTERACTION	(9)

Evaluation Techniques – Assessing user experience – Usability testing – Heuristic evaluation and walkthroughs – Analytics predictive models. Cognitive models – Socio-organizational issues and stakeholder requirements – Communication and collaboration models.

UNIT – IV	MODELS AND THEORIES	(9)

Task analysis – Dialog notations and design – Models of the system – Modeling rich interaction – Ubiquitous computing.

UNIT – V	WEB AND MOBILE INTERACTION	(9)

Hypertext – Multimedia and WWW – Designing for the web Direct Selection – Contextual Tools – Feedback patterns Mobile apps – Mobile navigation – Content and control idioms – Multi-touch gestures – Interapp integration – Mobile web.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Describe fundamental HCI concepts and user-centered design.	**Understand**
CO2	Recognize and explain various interaction styles	**Understand**
CO3	Use techniques to assess user experience and interactions.	**Apply**
CO4	Infer task analysis and interaction models	**Understand**
CO5	Design effective web and mobile interfaces with usability in mind	**Apply**

REFERENCES:

1. Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Niklas Elmqvist, "Designing the User Interface: Strategies for Effective Human-Computer Interaction", Pearson Education, India, 6th Edition, 2016.
2. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, "Human Computer Interaction", Pearson Education, New Delhi, 3rd Edition, 2004.
3. Helen Sharp Jennifer Preece Yvonne Rogers, "Interaction Design: Beyond Human Computer Interaction", Wiley, India, 5th Edition, 2019.
4. Alan Cooper, Robert Reimann, David Cronin, Christopher Noessel, "About Face: The Essentials of Interaction Design", Wiley, India, 4th Edition, 2014.
5. Wilbert O Galitz, "The Essential Guide to User Interface Design", Wiley India Pvt., Ltd, 3rd Edition, 2007.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	2	1	3	3	1
CO2	2	1	3	3	1
CO3	2	1	3	3	1
CO4	2	1	3	3	1
CO5	2	1	3	3	1
Avg.	**2**	**1**	**3**	**3**	**1**

1-low, 2-medium, 3-high

BD24E05	WEB SERVICES AND API DESIGN (PROFESSIONAL ELECTIVES – I and II)	Category	L	T	P	C
		PEC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Students should know web development basics, HTTP protocols, and XML/JSON formats. Familiarity with programming and object-oriented design is recommended and general software engineering principles.

OBJECTIVES:

1. To learn the architecture and design principles of web services and Service-Oriented Architecture
2. To study the building blocks of web services, including SOAP, WSDL, and UDDI.
3. To discover the fundamentals of RESTful web services, including HTTP methods and client interactions.
4. To gain hands-on experience with implementing RESTful web services using technologies like Spring and S3.
5. To learn to design and implement resource-oriented services, focusing on URIs, statelessness, and resource representation.

UNIT – I	BASICS OF WEB SERVICE	(9)

Overview – Web Service Architecture – Service-Oriented Architecture (SOA) – Architecting Web Services: Web Services Technology Stack – Logical Architectural View – Deployment Architectural View and Process Architectural View.

UNIT – II	WEB SERVICE BUILDING BLOCKS	(9)

Introduction to SOAP: SOAP Syntax – Sending SOAP Messages – SOAP Implementations – Introduction to WSDL: WSDL Syntax – SOAP Binding – WSDL Implementations – Introduction to UDDI: The UDDI API – Implementations –The Future of UDDI.

UNIT – III	RESTFUL WEB SERVICES	(9)

Programmable Web – HTTP: Documents in Envelopes – Method Information – Scoping Information – The Competing Architectures – Technologies on the Programmable Web – Leftover Terminology – Writing Web Service Clients: The Sample Application – Making the Request: HTTP Libraries – Processing the Response: XML Parsers.

UNIT – IV	IMPLEMENTATION OF RESTFUL WEB SERVICES	(9)

Introducing the Simple Storage Service – Object-Oriented Design of S3 – Resources – HTTP Response Codes Resource – URIs – Addressability – Statelessness – Representations – Links and Connectedness – The Uniform Interface – Spring Web Services – Spring MVC Components – Spring Web Flow – A Service Implementation using Spring Data REST.

UNIT – V	RESOURCE ORIENTED ARCHITECTURE	(9)

Resource – URIs – Addressability – Statelessness – Representations – Links and Connectedness – The Uniform Interface – Designing Read-Only Resource-Oriented Services: Resource Design – Turning Requirements into Read-Only Resources – Figure Out the Data Set – Split the Data Set into Resources – Name the Resources – Design Representation – Link the Resources to Each Other – The HTTP Response.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Describe web service architecture and SOA principles.	**Understand**
CO2	Identify and explain SOAP, WSDL, and UDDI components.	**Understand**
CO3	Apply RESTful principles to create and manage web services.	**Apply**
CO4	Implement RESTful services using frameworks like Spring and S3.	**Apply**
CO5	Develop and structure web services based on resource-oriented architecture principles.	**Apply**

REFERENCES:

1. Leonard Richardson and Sam Ruby, "RESTful Web Services", O'Reilly Media, 1st Edition, 2007
2. McGovern, et al., "Java Web Services Architecture", Morgan Kaufmann Publishers, 2nd Edition, 2005.
3. Lindsay Bassett, "Introduction to JavaScript Object Notation", O'Reilly Media, 2nd Edition, 2015
4. Craig Walls, "Spring in Action", Manning Publications, Shelter Island, 5th Edition, 2018
5. Raja CSP Raman, Ludovic Dewailly, "Building A RESTful Web Service with Spring 5", Packt Publishing, 2nd Edition,2018.
6. Bogunuva Mohanram Balachandar, "Restful Java Web Services: A pragmatic guide to designing and building RESTful APIs using Java", Ingram short title, 3rd Edition, 2017.
7. Mario-Leander Reimer, "Building RESTful Web Services with Java EE 8: Create modern RESTful web services with the Java EE 8 API", Packt publishing, 2nd Edition, 2018.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	1	2	3	1
CO2	3	1	2	3	1
CO3	3	1	2	3	1
CO4	3	1	2	3	1
CO5	3	1	2	3	1
Avg.	**3**	**1**	**2**	**3**	**1**

1-low, 2-medium, 3-high

BD24T27	MACHINE LEARNING TECHNIQUES (PROFESSIONAL ELECTIVES – III and IV)	Category	L	T	P	C
		PCC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

A fundamental understanding of mathematics, including probability and statistics, is essential. Basic programming skills, particularly in Python, are required. Familiarity with fundamental computer science concepts and data analysis techniques.

OBJECTIVES:

1. To provide a solid foundation in the core principles and mathematical foundations of machine learning.
2. To explore various supervised learning methods, including regression, classification, and neural networks.
3. To learn clustering and dimensionality reduction techniques in unsupervised learning.
4. To study the applications of graphical models like Bayesian Networks and Markov Random Fields.
5. To gain the concepts of reinforcement learning and sampling methods.

UNIT– I	BASICS OF MACHINE LEARNING	(9)

Machine Learning – Machine Learning Foundations – Overview – Design of a Learning System – Types of Machine learning – Applications Mathematical foundations of Machine Learning – Random Variables and Probabilities – Probability Theory – Probability Distributions – Decision Theory – Bayes Decision Theory – Information Theory.

UNIT – II	SUPERVISED LEARNING	(9)

Linear Models for Regression – Linear Models for Classification – Naive Bayes – Discriminant Functions – Probabilistic Generative Models – Probabilistic Discriminative Models – Bayesian Logistic Regression – Decision Trees – Classification Trees – Regression Trees – Pruning – Neural Networks – Feed Forward Network Functions – Back-Propagation – Support vector machines – Ensemble methods – Bagging – Boosting.

UNIT – III	UNSUPERVISED LEARNING	(9)

Clustering – K means – EM Algorithm – Mixtures of Gaussians – Curse of Dimensionality – Dimensionality Reduction – Factor Analysis – Principal Component Analysis – Probabilistic PCA.

UNIT – IV	PROBABILISTIC GRAPHICAL MODELS	(9)

Graphical Models – Undirected Graphical Models – Markov Random Fields – Directed Graphical Models – Bayesian Networks – Conditional Independence Properties – Inference – Generalization – Hidden Markov Models.

UNIT – V	ADVANCED LEARNING	(9)

Sampling – Basic Sampling methods – Monte Carlo. Reinforcement Learning – K-Armed Bandit – Elements – Model-Based Learning – Value Iteration – Policy Iteration –Temporal Difference Learning – Exploration Strategies.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Describe the fundamental concepts and mathematical foundations of machine learning.	**Understand**
CO2	Implement and apply various supervised learning models in real-world scenarios.	**Apply**
CO3	Recognize and apply unsupervised learning techniques like clustering and PCA.	**Understand**
CO4	Model and infer data using probabilistic graphical models.	**Understand**
CO5	Construct and implement advanced techniques like reinforcement learning.	**Apply**

REFERENCES:

1. Ethem Alpaydin, "Introduction to Machine Learning", MIT Press, London, 3rd Edition, 2014.
2. Kevin P. Murphy, "Machine Learning: A Probabilistic Perspective", MIT Press, London, 3rd Edition, 2012
3. Trevor Hastie, Robert Tibshirani and Jerome Friedman, "The Elements of Statistical Learning", Springer, United States, 2nd Edition, 2011.
4. Christopher Bishop, "Pattern Recognition and Machine Learning", Springer, United States, 1st Edition, 2007.
5. Tom M Mitchell, "Machine Learning", McGraw Hill Education, India, 1st Edition, 2013.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	1	3	2	1
CO2	3	1	3	2	1
CO3	3	1	3	2	1
CO4	3	1	2	2	1
CO5	3	1	3	2	1
Avg.	**3**	**1**	**3**	**2**	**1**

1-low, 2-medium, 3-high

CS24E08	SOFTWARE QUALITY ASSURANCE (PROFESSIONAL ELECTIVES – III and IV)	Category	L	T	P	C
		PCC	3	0	0	3

PREREQUISITE

Students should have a foundational understanding of software development processes and basic principles of software engineering. Familiarity with quality management concepts and practices is essential, as is an awareness of project management fundamentals. Previous experience with software testing and project lifecycle management will also be beneficial for grasping the integration of quality activities and the application of various quality assurance methodologies.

OBJECTIVES:

1. To learn the importance and factors of software quality and the components of software quality assurance
2. To Explore how to integrate quality activities throughout the software project lifecycle.
3. To Establish procedures and infrastructure for maintaining software quality, including training and configuration management.
4. To gain knowledge of various software quality metrics and cost models for effective quality management.
5. To study of quality management standards, certifications, and assessment methodologies.

UNIT – I	INTRODUCTION TO SOFTWARE QUALITY AND ARCHITECTURE	(9)

Need for Software quality – Software quality assurance (SQA) – Software quality factors- McCall's quality model – SQA system components – Pre project quality components – Development and quality plans

UNIT – II	SQA COMPONENTS AND PROJECT LIFE CYCLE	(9)

Integrating quality activities in the project life cycle – Reviews – Software Testing – Quality of software maintenance components – Quality assurance for external participant's contribution – CASE tools for software quality Management.

UNIT – III	SOFTWARE QUALITY INFRASTRUCTURE	(9)

Procedures and work instructions – Supporting quality devices – Staff training and certification – Corrective and preventive actions – Configuration management – Software change control – Configuration management audit – Documentation control.

UNIT – IV	SOFTWARE QUALITY MANAGEMENT AND METRICS	(9)

Project process control – Software quality metrics – Cost of software quality – Classical quality cost model – Extended model – Application and Problems in application of Cost model

UNIT – V	STANDARDS, CERTIFICATIONS AND ASSESSMENTS	(9)

Quality management standards – ISO 9001 and ISO 9000-3 – Capability Maturity Models – CMM and CMMI assessment methodologies – Bootstrap methodology – SPICE Project – SQA project process standards – Organization of Quality Assurance – Role of management in SQA – SQA units and other actors in SQA systems.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Comprehend key software quality factors and SQA components.	**Understand**
CO2	Illustrate quality activities and reviews within the project lifecycle.	**Understand**
CO3	Develop and manage quality infrastructure, including training and documentation.	**Understand**
CO4	Apply software quality metrics and cost models to evaluate and control project quality.	**Apply**
CO5	Assess and apply relevant quality standards and certification models, such as ISO 9001 and CMMI.	**Apply**

REFERENCES:

1. Daniel Galin, "Software Quality Assurance", Pearson Publication, India, 1st Edition, 2009.
2. Alan C. Gillies, "Software Quality: Theory and Management", International Thomson Computer Press, India, 2nd Edition, 2011.
3. Kshirasagar Naim and Priyadarshi Tripathy, "Software Testing and Quality Assurance Theory and Practice", John Wiley & Sons Inc., USA, 1st Edition, 2008
4. Mordechai Ben-Menachem, "Software Quality: Producing Practical Consistent Software", International Thompson Computer Press, India, 1st Edition, 2014.
5. Solis Tech, "Quality Assurance: Software Quality Assurance Made Easy", Create Space Independent Publishing, South Carolina, 1st Edition, 2016.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	2	2	3	3
CO2	3	2	2	3	3
CO3	3	2	2	3	3
CO4	3	2	2	3	3
CO5	3	2	2	3	3
Avg.	**3**	**2**	**2**	**3**	**3**

1-low, 2-medium, 3-high

CS24E09	FULL STACK WEB APPLICATION DEVELOPMENT (PROFESSIONAL ELECTIVES – III and IV)	Category	L	T	P	C
		PEC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Students should have a fundamental understanding of web development concepts and programming basics. Familiarity with core programming languages such as JavaScript is essential, as well as a basic grasp of HTML and CSS.

OBJECTIVES:

1. To gain proficiency in JavaScript and Type Script for developing dynamic web applications.
2. To build and manage Angular components and data binding effectively.
3. To learn Node.js for building scalable and efficient server-side applications.
4. To configure and use Express.js for streamlined web application development.
5. To manage and integrate MongoDB with Node.js for effective data handling and storage.

UNIT – I	JAVASCRIPT AND TYPESCRIPT LANGUAGE	(9)

Server-Side Web Applications – Client-Side Web Applications – Understanding JavaScript: Types – Working with Arrays – Working with Objects – Understanding JavaScript Object Inheritance – Adding Type Declarations for the JavaScript Package – Adding Commands – Persistently Storing. Typescript: Data Types – Classes – Interfaces – Modules – Enumerations and Generics – Constructors – Functions – Getters and Setters.

UNIT – II	ANGULAR	(9)

Angular CLI – Anatomy of a Component – Data Binding: One Way Data Binding –Two Way Data Binding – Event Handling – Angular Module System – Directives – Types of Directives – Accessing the DOM Events in Directives Accessing the DOM Properties in Directives – Component Class Lifecycle.

UNIT – III	NODE.js	(9)

Basics of Node JS – Installation – Working with Node packages – Using Node package manager – Creating a simple Node.js application – Using Events – Listeners –Timers – Callbacks – Handling Data I/O – Implementing HTTP services in Node.js – Implementing Socket Services in Node.js.

UNIT – IV	EXPRESS.Js	(9)

Express.js: How Express.js Works. Configuration, Settings and Environment Middleware – Body Parser – Cookie –Parser – Express-session – Response time – Template Engine – Parameters and Routing – Router Class – Request Object – Response Object – Error Handling.

UNIT – V	MONGODB	(9)

Understanding NoSQL and MongoDB – Building MongoDB Environment – Administering User Accounts – Configuring Access control – Administering databases – Managing collections – Connecting to MongoDB from Node.js – Understanding the Objects Used in the MongoDB Node.js Driver – Accessing and Manipulating Databases – Manipulating MongoDB Documents from Node.js.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Infer in-depth knowledge of key features and data types for efficient web development.	**Understand**
CO2	Develop and manage Angular applications with skill.	**Understand**
CO3	Utilize Node.js for server-side development and handling various server-side tasks and services.	**Apply**
CO4	Construct and set up web applications with Express.js.	**Apply**
CO5	Integrate MongoDB with Node.js to manage data effectively in real-world applications.	**Apply**

REFERENCES:

1. Brad Dayley, Brendan Dayley, Caleb Dayley, "Node.js, MongoDB and Angular Web Development", Addison-Wesley, United States, 2nd Edition, 2018.
2. Adam Freeman, "Essential Typescript, Apress, United States, 1st Edition, 2019.
3. Mark Clow, "Angular Projects", Apress, United States, 1st Edition,2018.
4. Azat Mardan, "Pro Express.js", Apress, United States, 1st Edition,2015.
5. Chris Northwood, "The Full Stack Developer: Your Essential Guide to the Everyday Skills Expected of a Modern Full Stack Web Developer", Apress, 1st Edition, 2018.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	2	3	3	3	2
CO2	2	3	3	3	2
CO3	2	3	3	3	2
CO4	2	3	3	3	2
CO5	2	3	3	3	2
Avg.	**2**	**3**	**3**	**3**	**2**

1-low, 2-medium, 3-high

CS24E10	DEEP LEARNING (PROFESSIONAL ELECTIVES – III and IV)	Category	L	T	P	C
		PEC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Students should have a strong foundation in programming, preferably in Python, which is widely used in the field. A solid understanding of mathematics, including linear algebra, calculus and probability is essential for grasping deep learning algorithms and techniques.

OBJECTIVES:

1. To gain foundational knowledge of deep learning concepts, algorithms, and data types.
2. To acquire skills in building and optimizing neural networks.
3. To develop expertise in designing and implementing CNNs for image processing and transfer learning.
4. To learn techniques for NLP using Recurrent Neural Networks (RNNs) and LSTM and GRU.
5. To study the techniques like Q-Learning, GANs, and autoencoders.

UNIT– I	BASICS OF DEEP LEARNING	(9)

Fundamentals about Deep Learning – Perception Learning Algorithms – Probabilistic modelling – Early Neural Networks – Different from Deep Learning and Machine Learning – Scalars – Vectors – Matrixes – Higher Dimensional Tensors – Manipulating Tensors – Vector Data – Time Series Data – Image Data – Video Data.

UNIT – II	NEURAL NETWORKS	(9)

Fundamentals of Neural Network – Building Blocks of Neural Network – Optimizers. Activation Functions – Loss Functions – Data Pre-processing for neural networks – Feature Engineering – Overfitting and Underfitting – Hyper parameters.

UNIT – III	CONVOLUTIONAL NEURAL NETWORK	(9)

Introduction of CNN – Linear Time Invariant – Image Processing Filtering – Building a convolutional neural network – Input Layers – Convolution Layers – Pooling Layers – Dense Layers – Backpropagation Through the Convolutional Layer – Filters and Feature Maps – Backpropagation Through the Pooling Layers –Transfer Learning with Image Data – Transfer Learning using Inception Oxford VGG Model – Google Inception Model.

UNIT – IV	NATURAL LANGUAGE PROCESSING USING RNN	(9)

NLP and its Toolkits – Language Modeling – Vector Space Model (VSM) – Continuous Bag of Words (CBOW) –Skip-Gram Model for Word Embedding – Global Vectors for Word Representation GloVe – Backpropagation Through Time – Bidirectional RNNs (BRNN) – Long Short-Term Memory (LSTM) – Bi-directional LSTM – Sequence-to-Sequence Models – Gated recurrent unit GRU.

UNIT – V	DEEP REINFORCEMENT AND UNSUPERVISED LEARNING	(9)

Deep Reinforcement Learning – Q-Learning – Deep Q-Network (DQN) – Policy Gradient Methods – Actor-Critic Algorithm – Autoencoding – Convolutional Auto Encoding – Variational Auto Encoding – Generative Adversarial Networks – Autoencoders for Feature Extraction – Auto Encoders for Classification – Denoising Autoencoders – Sparse Autoencoders.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Demonstrate proficiency in deep learning principles and the ability to manage various data types.	**Understand**
CO2	Build and fine-tune neural networks using various activation and loss functions.	**Understand**
CO3	Develop and use convolutional neural networks for image-related tasks and transfer learning.	**Apply**
CO4	Apply RNNs, LSTM, and GRU models for processing and modeling language data.	**Apply**
CO5	Implement reinforcement learning methods and unsupervised techniques for data analysis and generation.	**Apply**

REFERENCES:

1. Brad Dayley, Josh Patterson and Adam Gibson, “Deep Learning A Practitioner’s Approach”, O’Reilly Media, USA, 1st Edition, 2017.
2. Jojo Moolayil, “Learn Keras for Deep Neural Networks”, Apress, Canada, 1st Edition, 2018.
3. Vinita Silaparasetty, “Deep Learning Projects Using TensorFlow 2”, Apress, Canada, 1st Edition, 2020.
4. Francois Chollet, “Deep Learning with Python”, Manning Shelter Island, 2nd Edition,2017.
5. Santanu Pattanayak, “Pro Deep Learning with TensorFlow”, Apress, Canada, 1st Edition, 2017.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	2	2	3	3
CO2	3	2	2	3	3
CO3	3	2	2	3	3
CO4	3	2	2	3	3
CO5	3	2	2	3	3
Avg.	**3**	**2**	**2**	**3**	**3**

1-low, 2-medium, 3-high

CS24E11	NATURAL LANGUAGE PROCESSING (PROFESSIONAL ELECTIVES – III and IV)	Category	L	T	P	C
		PEC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Knowledge of fundamental linguistic concepts such as syntax, semantics, and morphology. Proficiency in Python or another programming language used in NLP and Basic understanding of machine learning principles and algorithms.

OBJECTIVES:

1. To gain foundational knowledge of natural language processing (NLP) and its components.
2. To study statistical methods for text classification and sequence labeling.
3. To develop skills in parsing and dependency parsing using contextual embeddings.
4. To learn semantic role labeling, word sense disambiguation, and information extraction techniques.
5. To review methods for discourse analysis, question answering, and dialogue system development.

UNIT – I	BASICS OF NLP	(9)

Natural Language Processing – Components – Basics of Linguistics and Probability and Statistics – Words –Tokenization – Morphology – Finite State Automata

UNIT – II	STATISTICAL NLP AND SEQUENCE LABELING	(9)

N-grams and Language models – Smoothing –Text classification – Naïve Bayes classifier – Evaluation – Vector Semantics – TF – IDF – Word2Vec- Evaluating Vector Models – Sequence Labeling – Part of Speech – Part of Speech Tagging – Named Entities – Named Entity Tagging.

UNIT – III	CONTEXTUAL EMBEDDING	(9)

Constituency – Context Free Grammar – Lexicalized Grammars – CKY Parsing – Earley's Algorithm – Evaluating Parsers – Partial Parsing – Dependency Relations – Dependency Parsing –-Transition Based – Graph Based.

UNIT – IV	COMPUTATIONAL SEMANTICS	(9)

Word Senses and WordNet – Word Sense Disambiguation – Semantic Role Labeling – Proposition Bank – FrameNet – Selectional Restrictions – Information Extraction – Template Filling.

UNIT – V	DISCOURSE ANALYSIS AND SPEECH PROCESSING	(9)

Discourse Coherence – Discourse Structure Parsing – Centering and Entity Based Coherence – Question Answering –Factoid Question Answering – Classical QA Models – Chatbots and Dialogue systems – Frame-based Dialogue Systems – Dialogue – State Architecture.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Demonstrate comprehension of NLP basics, linguistics, and tokenization techniques.	**Understand**
CO2	Implement statistical models for text classification, sequence labeling and vector semantics evaluation.	**Apply**
CO3	Develop and apply parsing techniques and contextual embeddings for sentence structure analysis.	**Apply**
CO4	Analyze and apply techniques for word sense disambiguation and semantic role labeling.	**Apply**
CO5	Determine and develop discourse analysis models and dialogue systems, including question answering and chatbots.	**Understand**

REFERENCES:

1. Daniel Jurafsky and James H.Martin, "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition", Prentice Hall, India, 2nd Edition, 2020
2. Jacob Eisenstein, "Natural Language Processing", MIT Press, USA, 1st Edition, 2019
3. Samuel Burns, "Natural Language Processing: A Quick Introduction to NLP with Python and NLTK", 1st Edition, 2019
4. Nitin Indurkhya,Fred J. Damerau, "Handbook of Natural Language Processing", Chapman & Hall/CRC: Machine Learning & Pattern Recognition, Hardcover, 2nd Edition, 2010
5. Deepti Chopra, Nisheeth Joshi, "Mastering Natural Language Processing with Python", Packt Publishing Limited, Mumbai, 1st Edition, 2016

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	3	2	3	2
CO2	3	3	2	3	2
CO3	3	3	2	3	2
CO4	3	3	2	3	2
CO5	3	3	2	3	2
Avg.	**3**	**3**	**2**	**3**	**2**

1-low, 2-medium, 3-high

BD24E09	BLOCKCHAIN TECHNOLOGIES (PROFESSIONAL ELECTIVES – III and IV)	Category	L	T	P	C
		PEC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

A foundational knowledge of cryptography and hashing algorithms is essential. Familiarity with basic concepts in distributed systems and databases will also be beneficial. Understanding the principles of digital currencies and financial transactions will further aid in grasping blockchain applications.

OBJECTIVES:

1. To learn the basics of blockchain technology and its applications.
2. To study the architecture and types of cryptocurrencies.
3. To explore mechanisms to prevent double spending.
4. To gain insights into Bitcoin's key concepts and practical use.
5. To acquire about Ethereum, smart contracts, and blockchain applications.

UNIT – I	**BASICS OF BLOCKCHAIN**	**(9)**

Introduction to Blockchain – How Blockchain works – Blockchain vs Bitcoin – Practical applications – Public and Private key basics – Pros and Cons of Blockchain – Myths about Bitcoin.

UNIT – II	**BLOCKCHAIN AND CRYPTOCURRENCIES**	**(9)**

Architecture – Versions – Variants – Use cases – Life use cases of blockchain – Blockchain Vs Shared Database – Introduction to cryptocurrencies –Types – Applications.

UNIT – III	**CONCEPT OF DOUBLE SPENDING**	**(9)**

Concept of Double Spending – Hashing – Mining – Proof of work. Introduction to Merkel tree – Privacy – Payment verification –Resolving Conflicts – Creation of Blocks.

UNIT– IV	**BITCOIN**	**(9)**

Introduction to Bitcoin – key concepts of Bitcoin – Merits and De Merits Fork and Segwits – Sending and Receiving bitcoins – Choosing bitcoin wallet – Converting Bitcoins to Fiat Currency.

UNIT – V	**ETHEREUM AND BLOCKCHAIN APPLICATION**	**(9)**

Introduction to Ethereum – Advantages and Disadvantages – Ethereum vs Bitcoin – Introduction to Smart contracts – usage – Application – Working principle – Law and Regulations. Application: Medical Record Management System, Domain Name Service and Future of Blockchain.

TOTAL: 45 PERIODS

COURSE OUTCOMES:

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

COs	Course Outcome	Cognitive Level
CO1	Comprehend blockchain fundamentals and its practical uses.	**Understand**
CO2	Apply knowledge of cryptocurrency architecture and types in real-world scenarios.	**Apply**
CO3	Identify methods for preventing double spending and their effectiveness.	**Understand**
CO4	Elucidate Bitcoin's advantages, disadvantages, and transaction processes.	**Apply**
CO5	Build applications using Ethereum and smart contracts, considering legal and regulatory aspects.	**Understand**

REFERENCES:

1. Bikramaditya Signal, Gautam Dhameja, Priyansu Sekhar Panda, "Beginning Blockchain: A Beginner's Guide to Building Blockchain Solutions", APress, USA, 1st Edition, 2018,
2. Bahga, Vijay Madisetti, "Blockchain Applications: A Hands-On Approach", Arshdeep Bahga, USA, 1st Edition, 2018,
3. Alan Wright, "Blockchain - Hardcover Version: Uncovering Blockchain Technology, Cryptocurrencies, Bitcoin and the Future of Money", House of Books, Manchester, 1st Edition, 2021.
4. Arvind Narayanan & Joseph Bonneau & Edward Felten & Andrew Miller & Steven Goldfeder, "Bitcoin and Cryptocurrency Technologies", Princeton University Press, USA,1st Edition, 2016.
5. Andreas M. Antonopoulos, "Mastering Bitcoin Unlocking Digital Cryptocurrencies", Oreilly, USA, 1st Edition, 2014.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	1	3	3	1
CO2	3	1	3	3	1
CO3	3	1	3	3	1
CO4	3	1	3	3	1
CO5	3	1	3	3	1
Avg.	**3**	**1**	**3**	**3**	**1**

1-low, 2-medium, 3-high

BD24E10	CYBER PHYSICAL SYSTEMS (PROFESSIONAL ELECTIVES – III and IV)	Category	L	T	P	C
		PEC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

Basic understanding of embedded systems and control theory, as well as knowledge of computer networking and communication protocols. Awareness with real-time operating systems and scheduling algorithms and system design is essential, along with a foundation in verification techniques and security concepts.

OBJECTIVES:

1. To introduce the emergence and significance of CPS.
2. To learn methods for ensuring safety and liveness in CPS.
3. To investigate different models and protocols for designing CPS.
4. To study foundational aspects like security and synchronization in CPS.
5. To explore practical CPS applications using tools like CyberSim, Matlab, and Simulink.

UNIT – I	BASICS OF CYBER-PHYSICAL SYSTEMS	(9)

Cyber-Physical Systems (CPS) – Emergence of CPS – Key Features of Cyber-Physical Systems – CPS Drivers –Synchronous Model: Reactive Components – Properties of Components – Composing Components – Designs – Asynchronous Model of CPS: Processes – Design Primitives – Coordination Protocols.

UNIT – II	CPS REQUIREMENTS	(9)

Safety Specifications: Specifications – Verifying Invariants – Enumerative Search – Symbolic Search – Liveness Requirements: Temporal Logic – Model Checking – Proving Liveness.

UNIT – III	CPS MODELS	(9)

Dynamical Systems: Continuous – Linear Systems –Time Models – Linear Systems – Designing Controllers – Analysis Techniques – Timed Model: Processes – Protocols – Automata – Hybrid Dynamical Models.

UNIT – IV	CPS FOUNDATIONS	(9)

Symbolic Synthesis for CPS – Security in CPS – Synchronization of CPS – Real-Time Scheduling for CPS.

UNIT – V	APPLICATIONS AND PLATFORMS	(9)

Medical CPS – CPS Built on Wireless Sensor Networks – CyberSim User Interface – iClebo Kobuki – iRobot Create –myRIO – Cybersim – Matlab toolboxes – Simulink.

TOTAL: 45 PERIODS

COURSE OUTCOMES:

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

COs	Course Outcome	Cognitive Level
CO1	Comprehend the key features and drivers of Cyber-Physical Systems (CPS).	**Understand**
CO2	Examine synchronous and asynchronous models in CPS design.	**Understand**
CO3	Apply safety and liveness requirements to verify CPS behavior.	**Apply**
CO4	Design and apply continuous, linear, and hybrid dynamical systems for CPS.	**Apply**
CO5	Explore real-time scheduling, security and synchronization in CPS.	**Understand**

REFERENCES:

1. Raj Rajkumar, Dionisio De Niz, and Mark Klein, "Cyber-Physical Systems", Addison Wesley Professional, USA, 2016
2. Rajeev Alur, "Principles of Cyber-Physical Systems", MIT Press, USA, 1st Edition, 2015.
3. Lee, Edward Ashford, and Sanjit Arunkumar Seshia, "Introduction to embedded systems: A cyber physical systems approach", 2nd Edition, 2017
4. André Platzer, "Logical Analysis of Hybrid Systems: Proving Theorems for Complex Dynamics", Springer, USA, 1st Edition, 2010.
5. Jean J. Labrosse, "Embedded Systems Building Blocks: Complete and Ready-To-Use Modules in C", The publisher, Paul Temme, 1st Edition, 2011.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	3	1	3	2	1
CO2	3	1	3	2	1
CO3	3	1	3	2	1
CO4	3	1	3	2	1
CO5	3	1	3	2	1
Avg.	**3**	**1**	**3**	**2**	**1**

1-low, 2-medium, 3-high

CS24E12	GPU COMPUTING (PROFESSIONAL ELECTIVES – III and IV)	Category	L	T	P	C
		PEC	3	0	0	3

PREREQUISITE

A strong foundation in parallel computing and an understanding of computer architecture. Familiarity with C/C++ programming is essential for writing CUDA and OpenCL code. Knowledge of memory management and resource handling is important for optimizing applications. Additionally, a basic understanding of concurrency and synchronization concepts is needed for efficient parallel programming.

OBJECTIVES:

1. To study the history and evolution of GPU-based supercomputing.
2. To learn the fundamentals of CUDA hardware and memory management.
3. To develop problem-solving skills in CUDA programming and optimization.
4. To provide a comprehensive understanding of OpenCL basics and programming models.
5. To study concurrency and memory models in OpenCL for heterogeneous systems.

UNIT – I	GPU ARCHITECTURE	(9)

History of Supercomputing – Understanding Parallelism with GPU – CUDA Hardware Overview – Threads, Blocks, Grids, Warps, Scheduling – Memory Handling with CUDA: Shared, Global, Constant, and Texture Memory.

UNIT – II	CUDA PROGRAMMING	(9)

Using CUDA – Multi GPU – Multi GPU Solutions – Optimizing CUDA Applications: Problem Decomposition – Memory Considerations – Transfers, Thread Usage – Resource Contentions.

UNIT – III	CUDA PROGRAMMING ISSUES	(9)

Common Problems: CUDA Error Handling – Parallel Programming Issues, Synchronization – Algorithmic Issues – Finding and Avoiding Errors.

UNIT – IV	OPENCL BASICS	(9)

OpenCL Standard – Platform Model – Execution Model – Programming Model – Memory Model – Basic OpenCL Examples.

UNIT – V	CONCURRENCY MODEL	(9)

Commands and Queuing Model – Native and Built-in Kernels – Device side Queuing – Host-side Memory Model – Device side Memory Model – Dissecting OpenCL on Heterogeneous System.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Infer the architecture and parallelism of GPUs.	**Understand**
CO2	Develop CUDA programs and optimize applications.	**Apply**
CO3	Identify and resolve common issues in CUDA programming.	**Understand**
CO4	Explore the fundamentals of the OpenCL standard and its models.	**Apply**
CO5	Examine concurrency models in heterogeneous systems through the use of OpenCL.	**Understand**

REFERENCES:

1. Shane Cook, "CUDA Programming: A Developer's Guide to Parallel Computing with GPUs (Applications of GPU Computing)", Morgan Kaufmann, USA, 1st Edition, 2012
2. David R. Kaeli, Perhaad Mistry, Dana Schaa, Dong Ping Zhang, "Heterogeneous computing with OpenCL", Morgan Kauffman, USA, 3rd Edition, 2015.
3. Nicholas Wilt, "CUDA Handbook: A Comprehensive Guide to GPU Programming", Addison Wesley, Delhi, 1st Edition, 2013.
4. Jason Sanders, Edward Kandrot, "CUDA by Example: An Introduction to General Purpose GPU Programming", Addison Wesley, Delhi, 1st Edition, 2010.
5. https://opencl.org/

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	1	3	3	2	1
CO2	1	3	3	2	1
CO3	1	3	3	2	1
CO4	1	3	3	2	1
CO5	1	3	3	2	1
Avg.	**1**	**3**	**3**	**2**	**1**

1-low, 2-medium, 3-high

CS24E13	QUANTUM COMPUTING (PROFESSIONAL ELECTIVES – III and IV)	Category	L	T	P	C
		PEC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

A strong foundation in linear algebra, probability theory, and complex numbers. Familiarity with basic quantum mechanics and classical computing concepts is essential. Programming knowledge in quantum computing frameworks like qiskit or cirq.

OBJECTIVES:

1. To study basic quantum mechanics and qubit systems.
2. To learn quantum state transformations and quantum gates.
3. To explore key quantum algorithms and their generalizations.
4. To study quantum error correction and fault-tolerant computation techniques.
5. To investigate quantum cryptography, teleportation, and quantum information protocols.

UNIT – I	QUANTUM BUILDING BLOCKS	(9)

The Quantum Mechanics of Photon Polarization – Single-Qubit Quantum Systems – Quantum State Spaces – Entangled States – Multiple-Qubit Systems – Measurement of Multiple-Qubit States – EPR Paradox and Bell's Theorem – Bloch sphere.

UNIT – II	QUANTUM STATE TRANSFORMATIONS	(9)

Unitary Transformations – Quantum Gates – Unitary Transformations as Quantum Circuits – Reversible Classical Computations to Quantum Computations – Language for Quantum Implementations.

UNIT – III	QUANTUM ALGORITHMS	(9)

Computing with Superpositions – Quantum Subroutines – Quantum Fourier Transformations – Shor's Algorithm and Generalizations – Grover's Algorithm and Generalizations.

UNIT – IV	ENTANGLED SUBSYSTEMS AND ROBUST QUANTUM COMPUTATION	(9)

Quantum Subsystems – Properties of Entangled States – Quantum Error Correction – Graph states and codes – CSS Codes – Stabilizer Codes – Fault Tolerance and Robust Quantum Computing.

UNIT – V	QUANTUM INFORMATION PROCESSING	(9)

Limitations of Quantum Computing – Alternatives to the Circuit Model of Quantum Computation – Quantum Protocols – Building Quantum – Computers, Simulating Quantum Systems, Bell states. Quantum teleportation – Quantum Cryptography – No cloning theorem.

TOTAL: 45 PERIODS

COURSE OUTCOMES:

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

COs	Course Outcome	Cognitive Level
CO1	Infer the fundamental concepts of quantum mechanics and qubit systems.	**Understand**
CO2	Apply quantum gates and unitary transformations to quantum circuits.	**Apply**
CO3	Implement quantum algorithms such as Shor's and Grover's algorithms.	**Apply**
CO4	Make use of quantum error correction methods and fault-tolerant quantum computing.	**Apply**
CO5	Discover quantum protocols, cryptography, and the no-cloning theorem in quantum information processing.	**Understand**

REFERENCES:

1. John Gribbin, "Computing with Quantum Cats: From Colossus to Qubits", Bantam Press, New York, 3rd Edition, 2021.
2. William (Chuck) Easttom, "Quantum Computing Fundamentals", Addison-Wesley Professional, USA, 1st Edition, 2021.
3. Parag Lala, "Quantum Computing", McGraw-Hill Education, India, 1st Edition, 2019
4. Eleanor Rieffel and Wolfgang Polak, "Quantum Computing A Gentle Introduction", MIT Press, USA, 1st Edition, 2011.
5. Nielsen M. A., "Quantum Computation and Quantum Information", Cambridge University Press, England, 1st Edition, 2002.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	1	2	3	2	1
CO2	1	2	3	2	1
CO3	1	2	3	2	1
CO4	1	2	3	2	1
CO5	1	2	3	2	1
Avg.	**1**	**2**	**3**	**2**	**1**

1-low, 2-medium, 3-high

BD24E12	INFORMATION RETRIEVAL TECHNIQUES (PROFESSIONAL ELECTIVES – III and IV)	Category	L	T	P	C
		PEC	3	0	0	3

(Common to M.E CSE and M.E BDA)

PREREQUISITE

A foundational knowledge of data structures and algorithms, especially related to search and sorting techniques. Familiarity with basic probability, statistics, and machine learning concepts is essential for understanding classification, clustering, and retrieval models. Knowledge of web technologies and experience in programming languages like Python or Java will be beneficial for implementing IR systems.

OBJECTIVES:

1. To learn the foundational concepts and practical challenges in Information Retrieval (IR).
2. To study and analyze different IR models and their practical applications.
3. To learn various indexing methods and strategies for efficient query processing.
4. To gain the principles of text classification and clustering using advanced algorithms.
5. To explore the web search process, link analysis, and multimedia IR methods for enhanced retrieval.

UNIT – I	MOTIVATION	(9)

Basic Concepts – Practical Issues – Retrieval Process – Architecture – Boolean Retrieval – Retrieval Evaluation – Open-Source IR Systems – History of Web Search – Web Characteristics – Impact of the web on IR – IR Versus Web Search – Components of a Search Engine.

UNIT – II	MODELING	(9)

Taxonomy and Characterization of IR Models – Boolean Model – Vector Model – Term Weighting – Scoring and Ranking – Language Models – Set Theoretic Models – Probabilistic Models – Algebraic Models – Structured Text Retrieval Models – Models for Browsing.

UNIT – III	INDEXING	(9)

Static and Dynamic Inverted Indices – Index Construction and Index Compression. Searching – Sequential Searching and Pattern Matching – Query Operations – Query Languages – Query Processing – Relevance Feedback and Query Expansion – Automatic Local and Global Analysis – Measuring Effectiveness and Efficiency.

UNIT – IV	CLASSIFICATION AND CLUSTERING	(9)

Text Classification and Naive Bayes – Vector Space Classification – Support Vector Machines and Machine Learning on Documents. Flat Clustering – Hierarchical Clustering – Matrix Decompositions and Latent Semantic Indexing – Fusion and Meta Learning.

UNIT – V	SEARCHING THE WEB	(9)

Searching the Web – Structure of the Web – IR and web search – Static and Dynamic Ranking – Web Crawling and Indexing – Link Analysis – XML Retrieval Multimedia IR: Models and Languages – Indexing and Searching Parallel and Distributed IR – Digital Libraries.

TOTAL: 45 PERIODS

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

COs	Course Outcome	Cognitive Level
CO1	Comprehend the architecture and process of Information Retrieval (IR) systems.	**Understand**
CO2	Apply different IR models like Boolean, Vector, and Probabilistic models.	**Apply**
CO3	Implement indexing techniques, including static and dynamic inverted indices.	**Apply**
CO4	Perform text classification and clustering using machine learning methods.	**Understand**
CO5	Discover web search processes, web crawling, and multimedia IR techniques.	**Understand**

REFERENCES:

1. Ricardo Baeza, Yates, Berthier Ribeiro and Neto, "Modern Information Retrieval: The concepts and Technology behind Search", ACM Press Books, New York, 2nd Edition, 2011.
2. Stefan Buttcher, Charles L. A. Clarke and Gordon V. Cormack, "Information Retrieval Implementing and Evaluating Search Engines", The MIT Press, Cambridge, Massachusetts London, England, 3rd Edition, 2010.
3. Christopher D. Manning, Prabhakar Raghavan and Hinrich Schutze, "Introduction to Information Retrieval", Cambridge University Press, US, 1st Edition, 2008.
4. Gerald J. Kowalski, Mark T. Maybury, "Information Storage and Retrieval Systems: Theory and Implementation", Springer, US, 6th Edition, 2013.

Mapping of COs with POs and PSOs

COs/ POs	PO1	PO2	PO3	PSO1	PSO2
CO1	1	1	1	3	2
CO2	1	1	1	3	2
CO3	1	1	1	3	2
CO4	1	1	1	3	2
CO5	1	1	1	3	2
Avg.	**1**	**1**	**1**	**3**	**2**

1-low, 2-medium, 3-high