SYLLABUS
FOR
MASTER OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF ENGINEERING
UNIVERSITY OF KASHMIR
SRINAGAR

(Applicable to Batch 2024 & Onwards)

PROGRAM LEARNING OUTCOMES (PLOs)

Program Learning Outcomes for M.Tech in Computer Science and Engineering:

1. Computing Knowledge: Apply the advanced knowledge of computer science, mathematics, and engineering fundamentals to solve complex problems in software engineering and related fields.

2. Problem Analysis: Identify, formulate, and analyze complex computing problems by reviewing research literature and using first principles of computer science, algorithms, and software engineering sciences to reach substantiated conclusions.

3. Design/Development of Solutions: Design innovative computer based solutions for complex computing problems, considering public ethics, health and safety, and cultural, societal, and environmental factors while meeting specified requirements.

4. Conduct Investigations of Complex Problems: Utilize advanced research-based knowledge and methods, including experimental design, data analysis, and information synthesis, to investigate and provide valid conclusions for complex computing problems.

5. Modern Tool Usage: Develop and apply advanced techniques, tools, and resources, including predictive modeling and modern software engineering and development tools, to complex computing activities with an understanding of their limitations.

6. The Software Engineer and Society: Assess societal, ethical, legal, and cultural issues using contextual knowledge and apply this understanding to professional software development practices, recognizing the associated responsibilities.

7. Environment and Sustainability: Evaluate the impact of professional software engineering solutions on society and the environment, demonstrating a commitment to sustainable development practices.

8. Ethics: Adhere to and promote ethical principles and professional responsibilities in software engineering practices, with a strong emphasis on maintaining privacy and data security.

9. Individual and Team Work: Work effectively as an individual, as well as a member or leader of diverse and multidisciplinary teams, demonstrating strong collaboration and leadership skills.

10. Communication: Communicate complex technical concepts and solutions effectively with peers, professionals, and the broader community, through well-organized reports, design documentation, presentations, and clear instructions.

11. Project Management and Finance: Apply advanced knowledge of software engineering and management principles to plan, manage, and execute projects efficiently, considering financial constraints and resource management in multidisciplinary environments.

12. Life-long Learning: Recognize and engage in continuous learning and professional development to keep up with the evolving technological landscape and advancements in computer science and engineering.

These outcomes ensure that graduates are equipped with the necessary skills and knowledge to excel in their professional careers and contribute effectively to the advancement of the computer science and engineering field.

COURSE STRUCTURE (CURRICULUM) OF M.TECH. CSE (EFFECTIVE FROM BATCH 2024)

1st Semester

S. No.	Course Code	Course Name	Hours			Credits	Marks		
			L	T	P		ISE	ESE	Total
1	MCSECMF124	Mathematical Foundations of Computer Science	4	-	-	4	28	72	100
2	MCSECDS124	Advanced Data Structures	4	-	-	4	28	72	100
3	MCSECRM124	Research Methodology and IPR	2	-	-	2	14	36	50
4	MCSEDXX124	Program Elective-I	4	-	-	4	28	72	100
5	MCSEDXX124	Program Elective-II	4	-	-	4	28	72	100
6	MCSEAXX124	Audit Course	2	-	-	0	-	-	-
7	MCSELDS124	Advanced Data Structures Lab	-	-	4	2	14	36	50
8	MCSELXX124	Program Elective-II Lab	-	-	4	2	14	36	50
		Total:	**28**			**22**	**154**	**396**	**550**

2nd Semester

S. No.	Course Code	Course Name	Hours			Credits	Marks		
			L	T	P		ISE	ESE	Total
1	MCSECAL224	Advanced Algorithms	4	-	-	4	28	72	100
2	MCSECSC224	Soft Computing	4	-	-	4	28	72	100
3	MCSECPS224	Mini Project with Seminar	-	-	4	2	14	36	50
4	MCSEDXX224	Program Elective-III	4	-	-	4	28	72	100
5	MCSEDXX224	Program Elective-IV	4	-	-	4	28	72	100
6	MCSEAXX224	Audit Course	2	-	-	0	-	-	-
7	MCSELAL224	Advanced Algorithms Lab	-	-	4	2	14	36	50
8	MCSELXX224	Program Elective-IV Lab	-	-	4	2	14	36	50
		Total:	**30**			**22**	**154**	**396**	**550**

3rd Semester

S. No.	Course Code	Course Name	Hours			Credits	Marks		
			L	T	P		ISE	ESE	Total
1	MCSEDXX324	Program Elective-V	3	-	-	3	21	54	75
2	MCSEOXX324	Open Elective	3	-	-	3	21	54	75
3	MCSELXX324	Program Elective-V Lab	-	-	4	2	14	36	50
4	MCSEPDI324	Dissertation-I/Industrial Project	6	-	16	14	98	252	350
		Total:	**32**			**22**	**154**	**396**	**550**

4th Semester

S. No.	Course Code	Course Name	Hours			Credits	Marks		
			L	T	P		ISE	ESE	Total
1	MCSEPDI424	Dissertation-II/Industrial Project	8	4	20	22	154	396	550
		Total:	**32**			**22**	**154**	**396**	**550**

Elective Baskets

Program Electives I and II	
MCSEDAA124	Data Science
MCSEDAB124	Distributed Systems
MCSEDAC124	Data Preparation and Analysis
MCSEDAD124	Recommender System
MCSEDAE124	Machine Learning
MCSEDAF124	Data Storage Technologies and Networks
MCSEDAG124	Digital Image Processing
MCSEDAH124	Digital Forensics
MCSEDAI124	Ethical Hacking
MCSEDAJ124	Malware Analysis & Reverse Engineering
MCSEDAK124	Secure Software Design and Enterprise Computing
MCSEDAL124	Biometrics
MCSEDAM124	Next Generation Networks
MCSEDAN124	Graph Theory
LAB	
MCSELAA124	Data Science Lab
MCSELAB124	Distributed Systems Lab
MCSELAC124	Data Preparation and Analysis Lab
MCSELAD124	Recommender System Lab
MCSELAE124	Machine Learning Lab
MCSELAF124	Data Storage Technologies and Networks Lab
MCSELAG124	Digital Image Processing Lab
MCSELAH124	Digital Forensics Lab
MCSELAI124	Ethical Hacking Lab
MCSELAJ124	Malware Analysis & Reverse Engineering Lab
MCSELAK124	Secure Software Design and Enterprise Computing Lab
MCSELAL124	Biometrics Lab
MCSELAM124	Next Generation Networks Lab
MCSELAN124	Graph Theory Lab

Program Electives III and IV	
MCSEDAA224	Data Visualization
MCSEDAB224	Big Data Analytics
MCSEDAC224	Data Warehouse and Data Mining
MCSEDAD224	Data Security and Access Control
MCSEDAE224	Web Analytics and Development
MCSEDAF224	Knowledge Discovery
MCSEDAG224	Introduction to Deep Learning
MCSEDAH224	Pattern Recognition
MCSEDAI224	Intrusion Detection
MCSEDAJ224	Data Encryption & Compression
MCSEDAK224	Steganography & Digital Watermarking
MCSEDAL224	Information Theory & Coding
MCSEDAM224	Security Assessment and Risk Analysis
MCSEDAN224	Secure Coding
MCSEDAO224	Network Security
LAB	
MCSELAA224	Data Visualization Lab
MCSELAB224	Big Data Analytics Lab
MCSELAC224	Data Warehouse and Data Mining Lab
MCSELAD224	Data Security and Access Control Lab

MCSELAE224	Web Analytics and Development Lab
MCSELAF224	Knowledge Discovery Lab
MCSELAG224	Introduction to Deep Learning Lab
MCSELAH224	Pattern Recognition Lab
MCSELAI224	Intrusion Detection Lab
MCSELAJ224	Data Encryption & Compression Lab
MCSELAK224	Steganography & Digital Watermarking Lab
MCSELAL224	Information Theory & Coding Lab
MCSELAM224	Security Assessment and Risk Analysis Lab
MCSELAN224	Secure Coding Lab
MCSELAO224	Network Security Lab

Program Electives V	
MCSEDAA324	Big Data Processing Frameworks
MCSEDAB324	Cloud Computing
MCSEDAC324	Distributed Databases
MCSEDAD324	Natural Language Processing
MCSEDAE324	Social Network Data Analytics
MCSEDAF324	GPU Computing
MCSEDAG324	Web Search & Information Retrieval
MCSEDAH324	Software Defined Networks
MCSEDAI324	Blockchain Technology
LAB	
MCSELAA324	Big Data Processing Frameworks Lab
MCSELAB324	Cloud Computing Lab
MCSELAC324	Distributed Databases Lab
MCSELAD324	Natural Language Processing Lab
MCSELAE324	Social Network Data Analytics Lab
MCSELAF324	GPU Computing Lab
MCSELAG324	Web Search & Information Retrieval Lab
MCSELAH324	Software Defined Networks Lab
MCSELAI324	Blockchain Technology Lab

Open Electives	
MCSEOBA324	Business Analytics
MCSEOIS324	Industrial Safety
MCSEOOR324	Operations Research
MCSEOCE324	Cost Management of Engineering Projects
MCSEOCM324	Composite Materials
MCSEOWE324	Waste to Energy
Audit Course 1	
MCSEARP124	English for Research Paper Writing
MCSEADM124	Disaster Management
MCSEATK124	Sanskrit for Technical Knowledge
MCSEAVE124	Value Education
Audit Course 2	
MCSEACI224	Constitution of India
MCSEAPS224	Pedagogy Studies
MCSEASM224	Stress Management by Yoga
MCSEAPD224	Personality Development through Life Enlightenment Skills

1st Semester

Course Title: Mathematical Foundations of Computer Science							
Course Code: MCSECMF124					Examination Scheme		
Total Number of Lecture Hours: 50					External	72	
					Internal	28	
Lecture (L)	4	Practical (P)	0	Tutorial (T)	0	Total Credits	4

Course Objectives

- *To understand the mathematical fundamentals that support a variety of computer science courses, including data mining, network protocols, machine learning, and bioinformatics, and to develop a logical foundation for modern IT techniques like programming language design and concurrency.*
- *To provide a solid foundation in probability theory, statistical distributions, and limit theorems for uncertainty modelling in computational problems.*
- *To explore combinatorics, graph theory, and algebraic structures and their applications in networks, cryptography, and algorithm design.*

Course Content	No. of Teaching Hours
UNIT 1	**12 Hrs**
Probability mass functions (PMF), probability density functions (PDF), cumulative distribution functions (CDF). Parametric families of distributions: Binomial, Poisson, Normal, Exponential distributions. Expected value, variance, higher-order moments, conditional expectation. Law of Large Numbers, Central Limit Theorem, Markov's inequality, Chebyshev's inequality	
UNIT 2	**12 Hrs**
Random samples and sampling distributions. Methods of Moments and Maximum Likelihood Estimation (MLE). Hypothesis testing: Null and alternative hypotheses, Type I and Type II errors, p-values. Linear regression models: Simple and multiple regression. Classification problems: Binary and multiclass classification basics. Principal Component Analysis (PCA) for dimensionality reduction.	
UNIT 3	**12 Hrs**
Permutations and combinations: With and without repetition. Pigeonhole principle, inclusion-exclusion principle. Graph theory fundamentals: Graph isomorphism, planar graphs, Kuratowski's theorem. Graph colouring, Hamiltonian circuits, Eulerian cycles. Trees and spanning trees, minimal spanning trees (Prim's and Kruskal's algorithms). Introduction to Groups, Rings, and Fields, Rings and Fields in cryptographic applications.	
UNIT 4	**14 Hrs**
Introduction to Markov chains: Transition matrices, classification of states, steady-state behaviour. Model assessment techniques: Cross-validation, metrics for model evaluation. Overfitting and regularization. Introduction to recent trends: Probabilistic modelling in bioinformatics, Stochastic models in distributed systems and network security, Soft computing using probabilistic and algebraic methods, Graph-based anomaly detection in cybersecurity	

Books:

1. Probability and Statistics with Reliability, Queuing, and Computer Science Applications by Kishor S. Trivedi, PHI Learning.
2. M. Mitzenmacher and E. Upfal, Probability and Computing: Randomized Algorithms and Probabilistic Analysis, Cambridge University Press.
3. Fundamentals of Probability and Statistics for Engineers by T.T. Soong, Wiley India,
4. John Vince, Foundation Mathematics for Computer Science, Springer.
5. Alan Tucker, Applied Combinatorics, Wiley.

Course Learning Outcomes

CLO1: Apply probability and distribution concepts to real-world problems.
CLO2: Perform statistical analysis, including estimation, hypothesis testing, and regression.
CLO3: Use combinatorics, graph theory, and algebraic structures in algorithms and security.
CLO4: Analyse Markov chains, evaluate models, and explore applications in emerging fields.

Level of CLO-PLO Mapping

CLOs	PLOs											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	3	2	3	2	-	-	-	-	1	-	-
2	3	3	3	3	2	-	-	-	-	1	-	-
3	2	2	3	2	2	2	1	2	2	2	2	-
4	3	3	2	3	2	-	-	-	-	1	-	2

Course Title: Advanced Data Structures												
Course Code: MCSECDS124								**Examination Scheme**				
Total Number of Lecture Hours: 48								**External**		72		
								Internal		28		
Lecture (L)		**Practical (P)**	**0**	**Tutorial (T)**		**0**	**Total Credits**		**4**			

Course Objectives: *The aim of this course is to provide a comprehensive understanding of basic data structures including their representations, operations, and applications. Implement and apply advanced data structures like skip lists, AVL trees, B- trees etc. To familiarize students with hashing, graph traversals, and dynamic programming techniques. Apply various data structures and algorithms to real-world problems such as text processing, pattern matching, and compression algorithms.*

Course Content	No. of Teaching Hours
UNIT 1	**12 Hrs**
Basic concepts overview: Stack, Representation of stack in memory, Operations on Stacks, Implementation of Stack using arrays and linked list, Applications of stacks. Queues, Representation of Queue in Memory, Operations on Queue, Implementation of Queue using arrays and linked list, Queues, Implementation using Arrays and Linked list. Skip Lists: Need for Randomizing Data Structures and Algorithms, Search and Update Operations on Skip Lists, Probabilistic Analysis of Skip Lists, Deterministic Skip Lists.	
UNIT 2	**12 Hrs**
Trees: Definitions, terminologies and properties, Binary tree representation, traversals and applications, Threaded binary trees, Binary Search Trees, AVL Trees, M-way Search Trees, B-trees, B*-trees. Graphs: Terminology, Graph representations, Traversal Techniques, Operations on Graphs, Applications of Graphs Trees: Binary Search Trees, AVL Trees, Red Black Trees, 2-3 Trees, B-Trees, Splay Trees.	
UNIT 3	**12 Hrs**
Dictionaries: Definition, Dictionary Abstract Data Type, Implementation of Dictionaries. Hashing: Review of Hashing, Hash Function, Collision Resolution Techniques in Hashing, Separate Chaining, Open Addressing, Linear Probing, Quadratic Probing, Double Hashing, Rehashing, Extendible Hashing.	
UNIT 4	**12 Hrs**
Text Processing: String Operations, Brute-Force Pattern Matching, The Boyer- Moore Algorithm, The Knuth-Morris-Pratt Algorithm, Standard Tries, Compressed Tries, Suffix Tries, The Huffman Coding Algorithm, The Longest Common Subsequence Problem (LCS), Applying Dynamic Programming to the LCS Problem.	

Recommended Books:

1. *Mark Allen Weiss, Data Structures &Algorithm Analysis in C++, 2nd Edition, Pearson, 2004.*
2. *M T Goodrich, Roberto Tamassia, Algorithm Design, John Wiley, 2002.*
3. *A. M. Tenenbaum, Y. Langsam, and M. J. Augenstein, Data Structures Using C and C++, Prentice Hall, 2/e, 1995*

Course Learning Outcomes:

1. *Understand and implement linear data structures such as stacks, queues, and skip lists, and analyze their operations and memory representations.*
2. *Apply various tree and graph structures (binary trees, AVL trees, B-trees, Red-Black trees) and perform efficient traversals, insertions, deletions, and searches.*
3. *Analyse and Optimize Hashing Techniques: Students will be able to apply different hashing techniques (e.g., separate chaining, linear probing, double hashing) to resolve collisions and optimize dictionary operations, while understanding their time and space complexities.*
4. *Apply Advanced Algorithms: Students will be able to apply advanced algorithms such as the Boyer-Moore and Knuth-Morris-Pratt pattern matching algorithms, Huffman coding, and the Longest Common Subsequence (LCS) problem, along with understanding their efficiency.*

Level of CLO-PLO Mapping

CLOs	PLOs											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	3	2	2	1	1	-	1	1	2	1	2
2	3	3	2	2	2	-	-	1	1	2	1	1
3	3	3	2	2	2	1	-	1	2	2	-	1
4	3	3	2	2	2	1	1	1	1	2	2	2

Course Title: Research Methodology and IPR							
Course Code: MCSECRM124					Examination Scheme		
Total Number of Lecture Hours: 48					External	36	
					Internal	14	
Lecture (L)	2	Practical (P)	0	Tutorial (T)	0	Total Credits	2

Course Objectives

1. *To understand the basic concepts and processes of research.*
2. *To identify appropriate research problems and parameters.*
3. *To develop skills for scientific report writing and technical communication.*
4. *To understand the importance of Intellectual Property Rights (IPR) and patent procedures.*

Course Content	No. of Teaching Hours
UNIT 1	**10 Hrs**

Introduction to Research Methodology

Meaning of research, objectives and motivation of research: Types of research: fundamental, applied, descriptive, analytical: Research process and formulation of research problem: Criteria for good research

UNIT 2	**12 Hrs**

Literature Review and Technical Writing

Searching for literature: digital libraries, journals, databases: Literature survey and review techniques: Technical writing: structure of a research paper, proposal, thesis, and report writing. Journal metrics, indexing, and their significance in defining the quality of a journal.

UNIT 3	**12 Hrs**

Plagiarism and Research Ethics: Objectivity and subjectivity in research, integrity in research, and respect for intellectual property. Definition of plagiarism and plagiarism detection tools. Role of referencing/bibliography in handling plagiarism. Ethical publishing practices and transparency in authorship. University Grants Commission's (UGC) policy for curbing plagiarism.

UNIT 4	**14 Hrs**

Intellectual Property Rights and Patents

Introduction to IPR: Patents, designs, trade secrets, and copyrights. Process of patenting, innovation, and technological research. International patenting under PCT and global cooperation in intellectual property. Patent rights and scope, licensing, and technology transfer

Books:

1. C.R. Kothari, *Research Methodology: Methods and Techniques*, New Age International.
2. Ranjit Kumar, *Research Methodology: A Step-by-Step Guide for Beginners*, Sage Publications.
3. Levine, Stephan, Krehbiel, Berenson, *Statistics for Managers*, Pearson Education.
4. Prabuddha Ganguli, *Intellectual Property Rights: Unleashing the Knowledge Economy*, Tata McGraw-Hill.

Course Learning Outcomes

After completing this course, the students will be able to:

1. *Understand the foundational concepts of research methodology*
2. *Identify research problems, formulate hypotheses, and design experiments.*
3. *Conduct a comprehensive literature review and write research reports.*
4. *Understand the role and importance of IPR in research and innovation.*

Level of CLO-PLO Mapping

CLOs	PLOs											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	3	2	2	1	2	-	2	-	2	2	3
2	3	3	2	2	1	2	-	2	1	2	2	3
3	2	3	3	2	2	2	-	3	2	3	2	3
4	1	2	2	1	2	3	3	3	-	2	2	3

Course Title: Advanced Data Structures Lab							
Course Code: MCSELDS124					Examination Scheme		
Total Number of Lecture Hours: 30					External	36	
					Internal	14	
Lecture (L)	0	Practical (P)	4	Tutorial (T)	0	Total Credits	2

Course Objectives

To introduce and implement the basic and advanced data structures. Application of various data structures and algorithms in real-world problems such as text processing, pattern matching, and compression algorithms.

List of Experiments

1. Stack Implementation
2. Queue Implementation
3. Skip List Operations
4. Dictionary Implementation
5. Hashing Techniques
6. Binary Tree Operations
7. Binary Search Tree (BST) Operations
8. AVL Tree Operations
9. Graph Representation and Traversal
10. B-trees and B Trees*
11. Pattern Matching Algorithms
12. Tries and Compressed Tries
13. Huffman Coding Algorithm
14. Longest Common Subsequence (LCS)
15. Graph Algorithms (Advanced)
16. Dynamic Programming (Advanced Algorithms)

This is only a suggested list of experiments/simulations. The instructor is encouraged to familiarize students with additional relevant exercises.

Course Learning Outcomes:

1. *To implement and manipulate fundamental data structures such as stacks, queues, trees, and graphs using arrays and linked lists, and perform standard operations like insertion, deletion, and traversal.*
2. *Students will demonstrate the ability to implement and apply advanced data structures such as AVL trees, B-trees, and skip lists, and solve complex traversal and path finding problems in graphs.*
3. *Implement various hashing techniques (e.g., linear probing, separate chaining) and implement dictionaries for efficient data access & storage, analyzing their performance in terms of time & space.*
4. *Develop Solutions Using Text Algorithms and Dynamic Programming Students will be able to apply pattern matching algorithms (e.g., KMP, Boyer-Moore), Huffman coding, and dynamic programming-based solutions (like LCS), understanding their real-world applications and efficiency.*

Level of CLO-PLO Mapping

CLOs	PLOs											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	3	2	2	2	-	-	-	1	1	-	2
2	3	3	3	2	3	1	-	-	1	1	1	2
3	3	3	2	2	2	1	-	1	1	1	1	2
4	3	2	3	2	2	-	-	1	1	1	-	3

2nd Semester

Course Title: Advanced Algorithms							
Course Code: MCSECAL224					Examination Scheme		
Total Number of Lecture Hours: 40					External	72	
					Internal	28	
Lecture (L)	4	Practical (P)	0	Tutorial (T)	0	Total Credits	4

Course Objectives

To develop the ability to design, analyze, and implement advanced algorithms for complex problems. Students will learn efficient techniques such as greedy methods, dynamic programming, graph algorithms, and linear programming, along with understanding NP-completeness and advanced computational models.

Course Content	No. of Teaching Hours
UNIT 1	**10 Hrs**
Sorting: Review of various sorting algorithms, topological sorting. Graph: Definitions and Elementary Algorithms: Shortest path by BFS, shortest path in edge-weighted case (Dijkstra's), depth-first search and computation of strongly connected components, emphasis on correctness proof of the algorithm and time/space analysis, example of amortized analysis.	
UNIT 2	**10 Hrs**
Introduction to greedy paradigm, algorithm to compute a maximum weight maximal independent set. Application to MST. Graph Matching: Algorithm to compute maximum matching. Characterization of maximum matching by augmenting paths, Edmond's Blossom algorithm to compute augmenting path. Flow Networks: Maxflow-mincut theorem, Ford-Fulkerson Method to compute maximum flow, Edmond-Karp maximum-flow algorithm.	
UNIT 3	**10 Hrs**
Shortest Path in Graphs: Floyd-Warshall algorithm and introduction to dynamic programming paradigm. More examples of dynamic programming. Modulo Representation of integers/polynomials: Chinese Remainder Theorem, Conversion between base-representation and modulo representation. Schonhage-Strassen Integer Multiplication algorithm.	
UNIT 4	**10 Hrs**
Linear Programming: Geometry of the feasibility region and Simplex algorithm NP-completeness: Examples, proof of NP-hardness and NP completeness. One or more of the following topics based on time and interest Approximation algorithms, Randomized Algorithms, Interior Point Method, Advanced Number Theoretic Algorithm.	

Recommended Books:

1. *Introduction to Algorithms by Cormen, Leiserson, Rivest, Stein.*
2. *The Design and Analysis of Computer Algorithms by Aho, Hopcroft, Ullman.*
3. *Algorithm Design by Kleinberg and Tardos.*
4. *H. S. Wilf, Algorithms and complexity, Prentice Hall.*

Course Learning Outcomes:

1. *Analyze and apply advanced sorting and graph algorithms with correctness proofs and time/space complexity analysis.*
2. *Apply greedy algorithms and graph matching techniques to solve optimization problems.*
3. *Solve complex problems using dynamic programming, including shortest paths and integer multiplication.*
4. *Apply Linear Programming techniques and understand their geometric interpretations and algorithmic implementations*

Level of CLO-PLO Mapping

CLOs	PLOs											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	3	3	2	3	2	2	1	2	1	1	2
2	3	3	3	2	3	2	2	1	2	1	1	2
3	3	3	3	2	3	2	2	1	2	1	1	2
4	3	3	3	2	3	2	2	1	2	1	1	2

Course Title: Soft Computing							
Course Code: MCSECSC224					Examination Scheme		
Total Number of Lecture Hours: 56					External	72	
					Internal	28	
Lecture (L)	4	Practical (P)	0	Tutorial (T)	0	Total Credits	4

Course Objectives:

- To introduce soft computing concepts & techniques & foster their abilities in designing appropriate technique for a given scenario.
- To implement soft computing based solutions for real-world problems.
- To give students knowledge of non-traditional technologies and fundamentals of artificial neural networks, fuzzy sets, fuzzy logic, genetic algorithms.
- To provide students a hand-on experience on MATLAB/Python to implement various strategies.

Course Content	No. of Teaching Hours
UNIT 1	**14 Hrs**

SOFT COMPUTING and FUZZY LOGIC: Soft Computing Constituents, Fuzzy Sets, Operations on Fuzzy Sets, Fuzzy Relations, Membership Functions: Fuzzy Rules and Fuzzy Reasoning, Fuzzy Inference Systems, Fuzzy Expert Systems, Fuzzy Decision Making.

UNIT 2	14 Hrs

NEURAL NETWORKS: Machine Learning Using Neural Network, Adaptive Networks, Feed forward Networks, Supervised Learning Neural Networks, Radial Basis Function Networks, Unsupervised Learning Neural Networks,.

UNIT 3	14 Hrs

DEEP LEARNING and GENETIC ALGORITHMS: Recent Trends in deep learning, various classifiers. Introduction to Genetic Algorithms (GA), Applications of GA in Machine Learning. Implementation of recently proposed soft computing techniques..

UNIT 4	14 Hrs

Matlab/Python Lib: Introduction to Matlab/Python, Arrays and array operations, Functions and Files, Study of machine learning/soft computing toolbox/libraries, Simple implementation of machine learning/soft computing techniques.

Recommended Books:

1. *Jyh:Shing Roger Jang, Chuen:Tsai Sun, EijiMizutani, Neuro:Fuzzy and Soft Computing , Prentice:Hall of India, 2003.*
2. *George J. Klir and Bo Yuan, Fuzzy Sets and Fuzzy Logic:Theory and Applications , Prentice Hall, 1995.*
3. *MATLAB Toolkit Manual*

Course Learning Outcomes:

After completion of course, students would be able to:

- *Identify and describe soft computing techniques and their roles in building intelligent machines*
- *Apply fuzzy logic and reasoning to handle uncertainty and solve various engineering problems.*
- *Apply genetic algorithms to combinatorial optimization problems.*
- *Evaluate and compare solutions by various soft computing approaches for a given problem*

Level of CLO-PLO Mapping

CLOs	PLOs											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	2	1	1	1	1	-	1	-	1	-	2
2	3	3	2	2	2	1	-	1	1	1	-	2
3	3	2	3	2	3	1	-	1	2	2	2	2
4	3	2	2	2	3	1	-	1	1	2	1	3

Course Title: Mini Project with Seminar												
Course Code: MCSECPS224								**Examination Scheme**				
Total Number of Lecture Hours: 30								**External**		36		
								Internal		14		
Lecture (L)	**0**	**Practical (P)**	**4**	**Tutorial (T)**		**0**	**Total Credits**					**2**

Course Objectives

1. To develop technical presentation and research communication skills.
2. To enhance the ability to review literature and identify relevant research areas.
3. To design and implement a mini project addressing a real-world or research-based problem.
4. To encourage innovation and application of theoretical knowledge to practical problems.

Description

Literature Survey & Problem Identification

Identifying a domain of interest
Surveying recent research papers, patents, and open problems
Defining scope and significance of the problem
Framing project objectives and deliverables

Design and Implementation

System architecture/design models
Choice of tools, algorithms, datasets, or simulations
Implementation phases: coding, testing, modelling
Iterative development and testing strategies

Documentation & Report Writing

Technical writing standards and structure
Preparation of interim and final reports
Citing references (IEEE/APA style)
Plagiarism checking and ethics in research

Seminar & Presentation

Preparing slides and poster presentations
Verbal and visual communication skills
Feedback-based refinement
Final seminar and viva-voce

Course Learning Outcomes: By the end of this course, students will be able to:

1. *Identify and articulate a research problem through comprehensive literature review.*
2. *Demonstrate the ability to design, model, or simulate a technical solution.*
3. *Present technical content effectively in oral and written form.*
4. *Collaborate in a team environment to complete a research-oriented mini project.*

Level of CLO-PLO Mapping

CLOs	PLOs											
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
1	3	3	2	2	1	1	1	1	2	2	2	3
2	3	2	3	2	3	0	1	1	2	2	2	2
3	2	1	1	1	1	0	0	0	1	3	3	2
4	2	2	2	2	2	1	1	1	3	3	-	-

Course Title: Advanced Algorithms Lab							
Course Code: MCSELAL224					Examination Scheme		
Total Number of Lecture Hours: 30					External	36	
					Internal	14	
Lecture (L)	0	Practical (P)	4	Tutorial (T)	0	Total Credits	2

Course Objectives

The aim of the Lab is to develop the ability to design, implement, and analyze advanced algorithmic techniques for solving complex computational problems, including graph algorithms, greedy strategies, dynamic programming, number-theoretic algorithms, and NP-complete problem approximations.

List of Experiments

1. Implement and analyze basic sorting algorithms (e.g., Merge Sort, Quick Sort, Heap Sort) with time and space complexity evaluation.
2. Implement BFS & DFS for traversal and shortest path computation in unweighted graphs.
3. Implement Dijkstra's algorithm for finding the shortest paths in weighted graphs.
4. Implement a Greedy algorithm for MST using Kruskal's and Prim's algorithms.
5. Implement graph matching algorithms (e.g., basic augmenting paths and Edmond's Blossom Algorithm for maximum matching).
6. Solve the Maximum Flow Problem using Ford-Fulkerson Method and Edmond-Karp Algorithm.
7. Implement Floyd-Warshall algorithm for finding all-pairs shortest paths.
8. Apply Chinese Remainder Theorem (CRT) for solving modular arithmetic problems.
9. Implement efficient Integer Multiplication using Schonhage-Strassen algorithm (basic version or an optimized large number multiplication).
10. Solve Linear Programming problems using the Simplex Method manually and/or using available libraries like SciPy. optimize. linprog.
11. Study and simulate an NP-complete problem (e.g., Vertex Cover or Subset Sum problem)
12. Implement an Approx. Algorithm for a classical NP-hard problem (e.g., Vertex Cover, TSP).
13. Implement a basic Randomized Algorithm (e.g., Randomized QuickSort or Randomized Selection Algorithm).

This is only a suggested list of experiments/simulations. The instructor is encouraged to familiarize students with additional relevant exercises.

Course Learning Outcomes:

1. *Implement sorting algorithms (Merge, Quick, Heap), graph algorithms (BFS, DFS, Dijkstra), and perform time-space complexity analysis.*
2. *Implementation of solutions for problems like MST (Kruskal's/Prim's), graph matching, maximum flow (Ford-Fulkerson, Edmond-Karp), and shortest paths (Floyd-Warshall).*
3. *Solve modular arithmetic and number-theoretic problems using efficient algorithms, apply Chinese Remainder Theorem, perform efficient integer multiplication (Schonhage-Strassen), and solve linear programming problems using Simplex Method.*
4. *Explore and simulate NP-complete problems, approximation, and randomized algorithms Students will investigate NP-hard problems (like Vertex Cover), implement approximation and randomized algorithms, and evaluate their computational complexity and performance.*

Level of CLO-PLO Mapping

CLOs	PLOs											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	3	2	2	2	-	-	-	1	1	-	2
2	3	3	3	2	2	-	-	-	1	1	-	2
3	3	3	2	2	2	1	-	1	-	1	-	3
4	3	3	2	2	1	-	-	-	1	1	1	3

3rd Semester

Course Title: Dissertation-I/ Industrial Project									
Course Code: MCSEPDI324						Examination Scheme			
Total Number of Lecture Hours: 30						External	252		
						Internal	98		
Lecture (L)	6	Practical (P)	16	Tutorial (T)	0	Total Credits	14		

Description

- In the Dissertation-I, students shall choose a specific topic/area for their dissertation and carry out the literature survey of the chosen area. Students are encouraged to work towards somereal-lifeproblem or issue/s of societal importance in order to ensure relevant research.Each student shall submit a dissertation report at the end of the third semester and appear in presentation/viva voce before the Departmental Committee. The dissertation report should also contain the problem specification and milestones to be achieved in solving theproblem.
- At the beginning of the third semester, a supervisor will be assigned to each student. TheSupervisor shall provide a syllabus and plan of study including relevant research papers tothe student. The student shall have to maintain a proper diary reflecting the activities and progress accomplished in his/her work and update the same regularly.
- The Supervisor shall monitor the progress of the student on weekly basis. Out of the 98 marks stipulated for Internal Semester Evaluation (ISE) of the Dissertation-I, fifty percent shall be awarded on the basis of continuous assessment by the respectiveSupervisor, while the remaining fifty percent shall be evaluated during the presentation/viva-voce to be held before the Departmental Committee.
- The External Semester Evaluation (ESE) shall be held by an approved external examiner. The External Semester Evaluation (ESE) shall be of 252 marks. The break-up of ESE 252 marks shall be as follows:

  Presentation: 20% marks
  Viva-voce: 40 % marks
  Dissertation writing based on state of art, fundamentals of the topic and its viability: 40 % marks

4th Semester

Course Title: Dissertation-II/ Industrial Project							
Course Code: MCSEPDI424					Examination Scheme		
Total Number of Lecture Hours: 30					External	396	
					Internal	154	
Lecture (L)	8	Practical (P)	20	Tutorial (T)	4	Total Credits	22

Description

- Dissertation-II shall commence with the fourth semester wherein a student accumulates 22 credits on successful completion of the same. This is in addition to the Dissertation-I duringthethird semester wherein a student shall choose a specific research topic/area andundertake its study.
- A thesis outlining the entire problem, including a survey of literature (results from Dissertation-I) and the various results obtained along with their solutions is expected tobe produced by each student. A Thesis Committee shall check the thesis for its completeness. A soft copy of the thesis in PDF format (in specific style) should be sent to the Thesis Committee, before its final submission. The Thesis Committee can recommend for modifications of the thesis or offer suggestions for improvement of the same for resubmission The Thesis committee shall also examine for suitability of publication (including any possible plagiarism) before the thesis goes in print and for binding.
- Consequent to the thesis being accepted and approved by the Thesis Committee, the Viva-voce examination of the student shall be conducted by an approved Examiner. Thecandidates who fail to submit the dissertation work within the stipulated time have to submit the same at the time of next ensuing examination.
- Out of the 154 marks stipulated for Internal Semester Evaluation (ISE) of the Dissertation-II, fifty percent shall be awarded on the basis of continuous assessment by the respective Supervisor, while the remaining fifty percent shall be evaluated during thepresentation/viva-voce to be held before the Departmental Committee. Out of the 396 marks stipulated for the External Semester Evaluation (ESE), fifty percent marks shall be awarded on the basis of viva-voce and fifty percent marks for general evaluation of thesis

PROGRAM ELECTIVE-I & II

Course Title: Data Science							
Course Code: MCSEDAA124					Examination Scheme		
Total Number of Lecture Hours: 48					External	72	
					Internal	28	
Lecture (L)	4	Practical (P)	0	Tutorial (T)	0	Total Credits	4

Course Objectives

- *To introduce the fundamental concepts of data science and its applications.*
- *To impart knowledge of statistical analysis, data preprocessing, and visualization.*
- *To develop skills for applying machine learning models on real-world data.*
- *To familiarize students with data science tools, techniques, and ethical aspects.*

Course Content	No. of Teaching Hours
UNIT 1	**11 Hrs**

Introduction to Data Science

What is Data Science? Need and importance, Data Science Life Cycle, Roles of a Data Scientist, Types of Data and Sources, Structured vs Unstructured Data, Introduction to Big Data and Hadoop Ecosystem

UNIT 2	**13 Hrs**

Data Preprocessing and Visualization

Data Cleaning: Handling missing data, outliers, Data Transformation: Normalization, encoding, Data Reduction Techniques, Exploratory Data Analysis (EDA), Visualization tools: Matplotlib, Seaborn, Tableau basics

UNIT 3	**12 Hrs**

Statistical Methods for Data Science

Descriptive and Inferential Statistics, Probability Distributions (Normal, Binomial, Poisson), Hypothesis Testing, Correlation and Regression Analysis, Sampling Methods and Estimation

UNIT 4	**12 Hrs**

Machine Learning Basics for Data Science

Introduction to Machine Learning, Supervised vs Unsupervised Learning, Regression, Classification, Clustering, Overfitting and Underfitting, Model Evaluation Metrics (Accuracy, Precision, Recall, F1 Score)

Recommended Books:

- *Joel Grus, "Data Science from Scratch", O'Reilly Media.*
- *Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow", O'Reilly Media.*
- *Chanchal Chatterjee, "Data Science and Analytics", McGraw Hill.*
- *Tirthajyoti Sarkar, "Data Science and Machine Learning Projects", Packt Publishing.*

Course Learning Outcomes:

1. *Understand the foundational concepts and processes of data science.*
2. *Perform data cleaning, transformation, and visualization tasks.*
3. *Apply statistical methods to analyze and interpret data.*
4. *Implement machine learning algorithms on datasets and evaluate models.*

Level of CLO-PLO Mapping

CLOs	PLOs											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	2	2	1	1	1	2	1	1	1	1	1
2	3	3	2	2	3	1	1	1	2	2	2	2
3	3	3	3	2	2	1	2	1	2	2	2	2
4	3	3	3	2	3	2	2	2	2	2	2	2

Course Title: Distributed Systems							
Course Code: MCSEDAB124					Examination Scheme		
Total Number of Lecture Hours: 48					External	72	
					Internal	28	
Lecture (L)	4	Practical (P)	0	Tutorial (T)	0	Total Credits	4

Course Objectives

- *To study foundations of distributed systems.*
- *To understand in detail network virtualization and remote invocations required for a distributed system.*
- *To introduce the idea of peer-to-peer services and file system.*
- *To understand clock synchronization techniques, transactions and concurrency control mechanisms.*

Course Content	No. of Teaching Hours
UNIT 1	**11 Hrs**
Introduction – Taxonomy of Distributed Systems - Scalable performance - load balancing and availability. Models of computation - shared memory and message passing system - synchronous and asynchronous systems. Various Paradigms in Distributed Applications.	
UNIT 2	**13 Hrs**
Communication in Distributed Systems- Remote Procedure Call – Remote Object Invocation-Message-Oriented Communication – Unicasting, Multicasting and Broadcasting – Group Communication. System Model – Inter process Communication - the API for internet protocols – External data representation and Multicast communication.	
UNIT 3	**12 Hrs**
Peer-to-peer Systems – Introduction - Napster and its legacy - Peer-to-peer – Middleware - Routing overlays. Distributed File Systems –Introduction - File service architecture – Andrew File system. Features-File model -File accessing models - File sharing semantics, Naming: Identifiers, Addresses, Name Resolution – Name Space Implementation – Name Caches – LDAP.	
UNIT 4	**12 Hrs**
Clocks, events and process states - Synchronizing physical clocks- Logical time and logical clocks - Global states – Coordination and Agreement – Introduction - Distributed mutual exclusion – Elections – Transactions and Concurrency Control– Transactions -Nested transactions – Locks – Optimistic concurrency control - Timestamp ordering – Atomic Commit protocols -Distributed deadlocks – Replication – Case study (Coda)	

Recommended Books:

- *Tanenbaum A.S., Van Steen M., "Distributed Systems: Principles and Paradigms", Pearson Education.*
- *George Coulouris, Jean Dollimore and Tim Kindberg, "Distributed Systems Concepts and Design", Fifth Edition, Pearson Education.*
- *Pradeep K Sinha, "Distributed Operating Systems: Concepts and Design", Prentice Hall of India.*
- *Liu M.L., "Distributed Computing, Principles and Applications", Pearson Education, 2004.*

Course Learning Outcomes:

1. *Understand the design principles and architecture of distributed systems.*
2. *Analyze the functioning of communication mechanisms such as RPC, multicasting, etc.*
3. *Analyze the design and functioning of existing distributed file systems.*
4. *Apply various distributed algorithms related to clock synchronization, concurrency control, deadlock detection, load balancing, etc.*

Level of CLO-PLO Mapping

CLOs	PLOs											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	2	1	1	1	–	–	–	–	–	–	2
2	3	3	2	2	2	–	–	–	–	1	–	2
3	2	3	2	2	2	–	–	–	–	1	–	2
4	3	3	3	2	3	–	–	–	–	1	1	2

Course Title: Data Preparation and Analysis							
Course Code: MCSEDAC124						Examination Scheme	
Total Number of Lecture Hours: 48						External	72
						Internal	28
Lecture (L)	4	Practical (P)	0	Tutorial (T)	0	Total Credits	4

Course Objectives

1. *To understand various types of data and their sources.*
2. *To learn techniques for data cleaning, transformation, integration, and reduction.*
3. *To prepare data for advanced analytics using feature engineering and selection.*
4. *To apply techniques for handling missing data, outliers, and noise in datasets.*

Course Content	No. of Teaching Hours
UNIT 1	**10 Hrs**

Introduction to Data and Data Sources

Types of data: structured, semi-structured, unstructured: Sources of data: databases, web, sensors, APIs: Understanding data formats: CSV, JSON, XML: Overview of data science workflow and the role of data preparation

UNIT 2	**12 Hrs**

Data Cleaning and Preprocessing

Handling missing data: deletion, imputation techniques: Identifying and treating outliers and noisy data: Data normalization and standardization: Data inconsistency detection and resolution

UNIT 3	**12 Hrs**

Data Integration, Transformation, and Reduction

Data integration from multiple sources: Schema integration and conflict resolution: Data transformation: aggregation, generalization, discretization, encoding: Data reduction: PCA, sampling, attribute subset selection

UNIT 4	**14 Hrs**

Feature Engineering and Data Preparation Tools

Feature extraction and construction: Feature selection: filter, wrapper, and embedded methods: Introduction to automated data preparation tools: Data preparation using Python: Pandas, NumPy, Scikit-learn

Books:

1. "Data Preparation for Data Mining" by Dorian Pyle
2. "Python for Data Analysis" by Wes McKinney
3. "Data Wrangling with Pandas" by Jacqueline Kazil & Katharine Jarmul
4. "Feature Engineering for Machine Learning" by Alice Zheng and Amanda Casari

Course Learning Outcomes

1. *Understand the types of data and their implications for data preparation.*
2. *Apply techniques for data cleaning, integration, transformation, and reduction.*
3. *Perform data preparation tasks using tools like Python and libraries like Pandas, NumPy, and Scikit-learn.*
4. *Evaluate and handle missing data, outliers, noise, and data inconsistencies.*

Level of CLO-PLO Mapping

CLOs	PLOs											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	2	1	1	1	-	-	-	-	1	-	-
2	3	3	2	2	2	-	-	-	-	2	-	-
3	3	3	3	2	3	2	-	-	1	2	1	2
4	2	3	3	2	2	1	-	-	-	2	-	1

Course Title: Recommender System							
Course Code: MCSEDAD124					Examination Scheme		
Total Number of Lecture Hours: 48					External	72	
					Internal	28	
Lecture (L)	4	Practical (P)	0	Tutorial (T)	0	Total Credits	4

Course Objectives

- *Students will be able to explain key concepts in information retrieval, including retrieval models, search techniques, and the role of relevance feedback and user profiles in enhancing search effectiveness.*
- *Students will develop skills in designing and implementing content-based filtering systems, including feature extraction, item profiling, and user profile learning methods.*
- *Students will evaluate different collaborative filtering approaches, including user-based and item-based methods, while understanding the challenges and potential vulnerabilities of these systems.*
- *Students will learn to assess the performance of various recommender systems using established evaluation metrics, and classify systems into categories*

Course Content	No. of Teaching Hours
UNIT 1	**12 Hrs**

Introduction to Recommender Systems (RS): Goals of RS, Basic models of RS, Challenges in RS. **Collaborative filtering:** Key properties of rating matrices, user and item based nearest recommendation, predicting ratings, neighbourhood-based methods (clustering, dimensionality reduction, regression modelling and graph models), Model based collaborative filtering, Content-based, knowledge based, ensemble based and hybrid recommender system.

UNIT 2	12 Hrs

Evaluating Recommender Systems: Explanations in recommender systems, General properties of evaluation research, popular evaluation designs, goals of evaluation design issues in offline recommender evaluation, accuracy metrics in offline evaluation. **Context, time and location sensitive RS:** Multidimensional approach, context pre filtering, post filtering, contextual modelling, temporal collaborative filtering, discrete temporal models, and location aware recommender systems.

UNIT 3	12 Hrs

Structural recommendations in networks Ranking algorithms, recommendations by collective classification, recommending friends: link prediction, social influence analysis and viral marketing.
Social and trust centric RS: Multidimensional models for social context, network centric and trust centric methods, user interaction in social recommenders.

UNIT 4	12 Hrs

Attack-resistant RS: Trade-offs Attack models, Types of attacks, detecting attacks on RS, strategies for robust RS, Online consumer decision making Learning to rank, multi-armed bandit algorithms, group RS, multi criteria RS, Active learning in RS, privacy in RS, Recommender systems and next generation web.

Recommended Books:

1. *Charu C. Aggarwal, Recommender Systems: The Textbook, Springer (2016), 1st ed.*
2. *Ricci F., Rokach L., Shapira D., Kantor B.P., Recommender Systems Handbook, Springer(2011), 1st ed.*
3. *Manouselis N., Drachsler H., Verbert K., Recommender Systems for Learning, Springer (2013), 1st ed.*

Course Learning Outcomes:

1. *Students will be able to describe the objectives of RS, differentiate among basic types and identify common challenges like data sparsity, scalability, and cold start problems.*
2. *Students will gain the ability to implement user-based and item-based nearest neighbour methods, clustering, dimensionality reduction, regression models, and hybrid approaches combining multiple recommendation techniques.*
3. *Students will learn to design offline evaluation experiments, utilize metrics such as precision, recall, MAE, and build context-aware, time-aware, and location-aware recommender systems.*
4. *Students will be able to model social influence, predict links, design robust RS against adversarial attacks, and explore current trends in next-gen web environments.*

Level of CLO-PLO Mapping

CLOs	PLOs											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	2	2	2	1	1	0	1	1	1	-	2
2	3	3	3	2	3	1	0	1	2	2	1	2
3	3	3	3	3	3	2	1	2	2	2	2	3
4	3	3	3	3	3	2	1	2	2	2	2	3