Name of Faculty	:	Faculty of Engineering & Technology
Name of Program	:	Master of Technology (M. Tech)
Course Code	:	2MSE04
Course Title	:	Advanced Data Structures & Algorithms (PE - II)
Type of Course	:	PE
Year of Introduction	:	2023-24

Prerequisite	:	Basic Data Structure, Basic Algorithm and OOP
Course Objective	:	The course objectives of an advanced data structures and algorithms course may vary depending on the institution or specific program. However, here are some common objectives you might find in such a course: Adv. Algorithm Techniques, Algorithm Analysis and Design, Complexity Theory etc.
Course Outcomes	:	At the end of this course, students will be able to:
	CO1	Design and analyse programming problem statements.
	CO2	Choose appropriate data structures and algorithms, understand the ADT/libraries, and use it to design algorithms for a specific problem.
	CO3	Understand the necessary mathematical abstraction to solve problems.
	CO4	Come up with analysis of efficiency and proofs of correctness.
	CO5	Comprehend and select algorithm design approaches in a problem specific manner.

Teaching and Examination Scheme

Teaching Scheme (Contact Hours)			**Credits**	**Examination Marks**				
				Theory Marks		**Practical Marks**		**Total Marks**
L	**T**	**P**	**C**	**SEE**	**CIA**	**SEE**	**CIA**	
4	0	2	5	70	30	30	20	100

Legends: ***L****-Lecture;* ***T****–Tutorial/Teacher Guided Theory Practice;* ***P*** *– Practical,* ***C*** *– Credit,* ***SEE*** *– Semester End Examination,* ***CIA*** *- Continuous Internal Assessment (It consists of Assignments/Seminars/Presentations/MCQ Tests, etc.))*

Course Content

Unit No.	Topics	Teaching Hours	Weightage	Mapping with CO
1	**Analysis of Algorithms** The efficiency of algorithm, average and worst-case analysis, elementary operation. Asymptotic Notation, analyzing control statement, Analyzing Algorithm using Barometer. Amortized analysis, solving recurrence Equation, Sorting Algorithm, Binary Tree Search.	08	08%	CO1
2	**Randomized Algorithms** Probability, Analyzing Quick Sort. Quick Select – median selection in linear time. Quick Sort and Treaps with High Probability, Treaps.	13	22%	CO2
3	**Graph Algorithms** Breadth First Search (BFS), Depth First Search (DFS). Topological Sort Strongly Connected Components, Euler Tour. Generic Minimum Spanning Tree, Kruskal's Algorithm, Prim's Algorithm, Single Source Shortest Path, Dijkstra's Algorithm, Bellman-Ford Algorithm.	11	18%	CO4
4	**String Matching** Introduction, The naïve string-matching algorithm. The Rabin-Karp algorithm, Knuth- Morris- Pratt Algorithm, Boyer- Moore Algorithm.	09	16%	CO3
5	**Approximation Algorithms** Greedy algorithms and approximation algorithms, Travelling Salesman Person, Approximation Algorithms for Set Cover and Clustering.	11	18%	CO5
6	**Computational Complexity** Introduction, Complexity classes, More NP-Complete problems Max-Clique, Independent Set, Vertex Cover, Graph Coloring, Hamiltonian Cycle and Travelling Salesman Problem.	08	11%	CO5

Suggested Distribution of Theory Marks Using Bloom's Taxonomy						
Level	Remembrance	Understanding	Application	Analyse	Evaluate	Create
Weightage	**40**	**20**	**20**	**10**	**-**	**10**

NOTE: This specification table shall be treated as a general guideline for the students and the teachers. The actual distribution of marks in the question paper may vary slightly from above table.

Suggested List of Experiments/Tutorials

Sr. No.	Name of Experiment/Tutorial (Any 12)	Teaching Hours
1	Write a program which creates Binary Search Tree. And also implement recursive and non-recursive tree traversing methods inorder, preorder and post-order for the BST.	01
2	Write a program to implement any two hashing methods. Use any one of the hashing methods to implement Insert, Delete and Search operations for Hash Table Management.	01
3	Explain Dictionary as an Abstract Data Type. Implement Dictionary using suitable Data Structure.	01
4	Write a program which creates AVLTree. Implement Insert and Delete Operations in AVL Tree. Note that each time the tree must be balanced.	01
5	Implement Red-Black Tree	02
6	Implement 2-3 Tree	02
7	Implement B Tree	01
8	Implement a program for String Matching using Boyer-Moore Algorithm on a text file content	02
9	Implement a program for String Matching using Knuth-Morris-Pratt Algorithm on a text file content.	01
10	Implement Huffman-Coding Method. Show the result with suitable example.	02
11	Implement Longest Common Subsequence (LCS) Problem using Dynamic Programming Method. Show the DP table and also find the particular solution of given strings.	01
12	Implement One Dimensional and Two-Dimensional Range Searching in any language.	01
13	Write a program which creates Priority Search Tree. Implement Insert and Search Operations in this Tree.	02
14	Write a program which creates Skip Lists. Implement Insert, Search and Update Operations in Skip-Lists.	01
15	Design a simple search engine to display the possible websites upon entering a search query. Use suitable data structure for storage and retrieval.	02
16	Prepare a Report/Presentation on Recent trends on Hashing/Trees/Computational Geometry to solve ay of recent evolving problem in real world.	01

Suggested Learning Websites

Sr. No.	Name of Website
1	http://www.cse.iitd.ernet.in/~naveen/courses/CSL630/sariel.pdf
2	http://www.cse.iitd.ernet.in/~naveen/courses/CSL630/jeff.pdf

Reference Books

Sr. No.	Name of Reference Books
1	Algorithm Design - Foundations, Analysis & Internet Examples by Michael T. Goodrich and Roberto Tamassia
2	Data Structures and Algorithms in Java by Michael T. Goodrich and Roberto Tamassia
3	Data Structures and Algorithms in C++ by Michael T. Goodrich, Roberto Tamassia and David M. Mount
4	Fundamental of Computer Algorithms by Ellis Horowitz, Sartaz sahni and sanguthevar Rajasekarm.