Design and Analysis of Algorithms

Course Code: **MCA- 201**	**L T C**
Course Name: **Design and Analysis of Algorithms**	**3 1 4**

INSTRUCTIONS TO PAPER SETTERS:
1. Question No. 1 should be compulsory and cover the entire syllabus. There should be 10 questions of short answer type of 2.5 marks each, having at least 2 questions from each unit.
2. Apart from Question No. 1, rest of the paper shall consist of four units as per the syllabus. Every unit should have two questions to evaluate the analytical/technical skills of the candidate. However, students may be asked to attempt only 1 question from each unit. Each question should be of 12.5 marks, including its subparts, if any.
3. Examiners are requested to go through the Course Outcomes (CO) of this course and prepare the question paper accordingly, using Bloom's Taxonomy (BT), in such a way that every question be mapped to some or other CO and all the questions, put together, must be able to achieve the mapping to all the CO(s), in balanced way.

LEARNING OBJECTIVES:
In this course, the learners will be able to develop expertise related to the following:-
1. Understand the important concepts of algorithms design and their analysis.
2. Analyze the efficiency of alternative algorithmic solutions to the problem.
3. Understand different algorithm paradigms like Divide and Conquer, Greedy, Dynamic, Backtracking and Branch and Bound.
4. Identify the appropriate data structures, algorithm design techniques and assess their impact on the performance of programs.

PRE-REQUISITES:
1. Programming Skills
2. Discrete Structures
3. Data Structures

COURSE OUTCOMES (COs):
After completion of this course, the learners will be able to:-

CO #	Detailed Statement of the CO	BT Level	Mapping to PO #
CO1	Demonstrate P and NP complexity classes of the problem.	BTL2	PO1, PO2, PO3
CO2	Apply the concepts of asymptotic notations to analyze the complexities of various algorithms.	BTL4	PO1, PO2, PO3, PO4
CO3	Analyze and evaluate the searching, sorting and tree-based algorithms.	BTL5	PO1, PO2, PO3, PO4, PO5
CO4	Design efficient solutions using various algorithms for given problems.	BTL6	PO1, PO2, PO3, PO4, PO5, PO6, PO10
CO5	Develop innovative solutions for real-world problems using different paradigms.	BTL6	PO1, PO2, PO3, PO4, PO5, PO6, PO7, PO9, PO10,

			PO11, PO12

UNIT – I

No. of Hours: 09 Chapter/Book Reference: TB1 [Chapters 1-5]; TB2 [Chapters 0-2]; TB3 [Chapters 2, 5, 13]

Performance Analysis of Algorithms: Algorithm Specification, Performance Analysis: Space and Time Complexity, Correctness of Algorithms, Growth of Functions, Asymptotic Notations and Types, Concept of Randomized Algorithms.
Recurrences: Substitution, Iteration, Master and Recurrence Tree method.

UNIT – II

No. of Hours: 12 Chapter/Book Reference: TB1 [Chapters 7-9, 13, 21 28, 32]; TB2 [Chapter 2]; TB3 [Chapter 5]

Divide and Conquer Paradigm: Problem Solving, Comparative Analysis of different Sorting and Searching Techniques, Strassen's Matrix Multiplication Method.
Sorting in linear time: Counting Sort, Bucket Sort and Radix Sort.
String Matching Concept: Naive String-Matching Algorithm, String Matching with Finite Automata, Knuth Morris Pratt Algorithm, The Rabin-Karp Algorithm.
Red Black Trees, Disjoint Set and their Implementation, Medians and Order Statistics.

UNIT – III

No. of Hours: 12 Chapter/Book Reference: TB1 [Chapters 15-16 & 23-25]; TB2 [Chapters 4-6]; TB3 [Chapters 4, 6]

Greedy Algorithms: General Concept, Applications, Activity Selection Problem, Fractional Knapsack problem, Job Sequencing with Deadlines, Huffman Coding, Analysis and Correctness of Prim's, Kruskal Algorithm and Dijkstra Algorithm.
Dynamic Programming: General Concept, Matrix-Chain Multiplication Problem, Longest Common Subsequence Problem, Bellman-Ford Algorithm, Analysis and Correctness of Floyd-Warshall Algorithm, Optimal Binary Search Trees, 0/1 Knapsack Problem, Network Flow Problem.

UNIT – IV

No. of Hours: 12 Chapter/Book Reference: TB1 [Chapters 34, 35]; TB2 [Chapters 8, 9]; TB3 [Chapter 8]

Backtracking: n-Queen's Problem, Hamiltonian Circuit Problem, Subset-Sum Problem, Graph Coloring Problem.
Branch and Bound: Assignment Problem, Travelling Salesman Problem.
Introduction to Computability, Polynomial-time Verification, NP-Completeness.
Complexity Classes: Reducibility, NP-Completeness Proof, NP-Complete & NP-Hard, Problem Classification-P, NP, NPC, NP-Hard; Circuit Satisfiability, 3SAT, Vertex Cover, Clique, Cook's Theorem.

TEXT BOOKS:

TB1. T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, "Introduction to Algorithms", PHI, 2nd Edition, 2006.
TB2. S. Dasgupta, C. Papadimitriou and U.Vazirani, "Algorithms", McGraw Hill Higher Education, 1st Edition, 2017.

TB3. J. Kleinberg and E. Tardos, “Algorithm Design”, Pearson Education, 2nd Edition, 2009.

REFERENCE BOOKS:

RB1. S. Horowitz, “Fundamentals of Computer Algorithms”, University Press, 2nd Edition, 2008.
RB2. R. Panneerselvam, “Design and Analysis of Algorithms”, PHI, 2nd Edition, 2016.
RB3. T. H. Cormen, “Algorithms Unlocked”, MIT Press, 1st Edition, 2013.
RB4. S. Sridhar, “Design and Analysis of Algorithms”, Oxford University Press, 1st Edition, 2014.
RB5. R. Neapolitan and K. Naimipour, “Foundations of Algorithms”, Jones & Bartlett Publishers, 4th Edition, 2010.
RB6. A. Levitin, “Introduction to the Design and Analysis of Algorithms”, Pearson Education, 3rd Edition, 2012.