C

Roll No. ..... Total Pages : 3

46216

BT-6/M-24

ANALYSIS & DESIGN OF ALGORITHMS

Paper– PE-IT-S310A

Time : Three Hours] [Maximum Marks : 75

Note : Attempt *five* questions in all, selecting at least *one* question from each unit. All questions carry equal marks.

UNIT–I

1. (a) Differentiate between big O, big Omega, and big Theta notation. How do you analyze the space complexity of an algorithm using asymptotic notation? Discuss.

   (b) Discuss the Divide and Conquer algorithmic paradigm, outlining its key steps and providing examples of problems that can be solved using this approach.

2. (a) Explain the Strassen matrix multiplication algorithm and discuss its significance in terms of time complexity compared to the conventional matrix multiplication algorithm.

   (b) Discuss quick sort and the worst-case time complexity of Quick Sort and explain the scenarios where it occurs?

UNIT–II

3. (a) Discuss the principles of Dynamic Programming with examples of problems that can be solved optimally using this approach.

   (b) Explain the concept of Minimum Cost Spanning Tree (MCST). Describe Kruskal's algorithm for finding the Minimum Cost Spanning Tree.

4. (a) Discuss the principles of the Greedy Method with examples of problems that can be solved optimally using this approach.

   (b) Provide a step-by-step example illustrating the application of Dynamic Programming to solve a small instance of the Travelling Salesman Problem.

UNIT–III

5. (a) Explain the Knapsack Problem and how it can be solved using the backtracking algorithmic technique.

   (b) Explain how a FIFO (First-In-First-Out) queue can be integrated into the Branch and Bound algorithm to manage the search space efficiently.

6. (a) Describe the Branch and Bound algorithmic technique and how it can be applied to solve the Traveling Salesman Problem.

   (b) Discuss the steps involved in solving the 8-Queen Problem using backtracking.

UNIT–IV

7. (a) Explain the concept of a Binary Search Tree (BST). Discuss how the BST property is maintained during insertion and deletion operations.

   (b) Provide examples of problems that belong to the NP-hard class but are not NP-complete. Explain why these problems are classified as NP-hard but do not meet the criteria for NP-completeness.

8. (a) Explain the structure and properties of a B+ tree. Discuss how B+ trees differ from B-trees.

   (b) Compare and contrast Depth-First Search (DFS) and Breadth-First Search (BFS) algorithms in terms of their traversal order, space complexity, and applications.