

2316

40

Class-M.Sc CS II Sem.

Subject - Design Analysis and Algorithms
Paper-MCS 203

Time Allowed : 3 Hours

Maximum Marks : 100

Section - A

Note:- (i) No extra sheet will be provided.

(ii) Attempt any five questions.

(iii) All questions carry equal marks.

1. What is an Algorithm? What is Algorithm specification? Explain it with pseudo code conventions. Also explain time and space complexity. 20

2. Write the general method for greedy algorithm design approach. Obtain an optimal solution to knapsack problem using greedy design approach for the following data:

No of items (n) = 7, Knapsack capacity (M) = 15,

Profits : $p(1) = 10$, $p(2) = 5$, $p(3) = 15$, $p(4) = 7$, $p(5) = 6$,
 $p(6) = 18$, $p(7) = 3$ and

Weights : $w(1) = 2$, $w(2) = 3$, $w(3) = 5$, $w(4) = 7$, $w(5) = 1$,
 $w(6) = 4$, $w(7) = 1$. 20

3. (a) Describe the general characteristics of problems, for which divide and conquer is unsuitable.

(b) Illustrate the operations of merge sort algorithm on the array and also compute the time complexity using divide and conquer strategy.

$A = \{310, 285, 179, 652, 351, 423, 861, 254, 450, 520\}$ 10+10=20

4. (a) Write Pseudo Code/Algorithm for Quick Sort technique. Explain it with Example. What is the complexity of this technique?
(b) Explain travelling salesman problem. How it can be solved using Dynamic Programming?
10+10=20
5. (a) Show that preorder and postorder sequences of a binary tree do not uniquely define the binary tree.
(b) Write down the algorithm to count the number of leaf nodes in a binary tree. Also calculate its time complexity.
10+10=20
6. Explain following terms in context of Backtracking algorithm design approach using 8-queens problem as an example:
(i) State Space (ii) Bounding Functions
(iii) Problem State (iv) Solution State(s) 20
7. Write algorithms for Breadth First Search (BFS) and Depth First Search (DFS) techniques. Also explain them with help of an example. 20
8. Discuss the following problems in context of dynamic programming:
(a) 0/1 Knapsack problem
(b) Multistage Graphs 20
