18CS42

Fourth Semester B.E. Degree Examination, July/August 2022 Design and Analysis of Algorithms

Max. Marks: 100 Time: 3 hrs.

Note: Answer any FIVE full questions, choosing ONE full question from each module.

Module-1

- Give the definition of an Algorithm and also discuss the characteristics of an Algorithm. 1 (05 Marks)
 - b. Define Space Complexity and Time Complexity of an algorithm and compute the time complexity of Fibbonocci Numbers algorithm. (05 Marks)
 - c. What are the various basic Asymptotic efficiency classes? Explain Big O, Big - Ω , Big - θ notations with examples. (10 Marks)

a. Give the Mathematical Analysis of Non recursive Matrix Multiplication Algorithm. 2

(05 Marks)

- b. Give the general plan for analyzing Time efficiency of Recursive algorithms and also Analyze the Tower of Hanoi Recursive algorithm. (10 Marks)
- c. Mention the important problem types considered for design and analysis. Explain any two problem types. (05 Marks)

Module-2

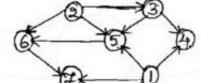
- a. Give the Recursive algorithm to find maximum and minimum element from the list and apply the algorithm to find maximum and minimum to the list [31, 22, 12, -7, 75, -6, 17, 47, 60]. (10 Marks)
 - b. Apply both mergesort and quicksort algorithm to sort the characters VTUBELAGAVI. (10 Marks)

a. Apply Strassen's algorithm for matrix multiplication to multiply the following matrices and justify how the Strassen's algorithm is better.

$$\begin{bmatrix} 4 & 3 \\ 1 & 2 \end{bmatrix} \times \begin{bmatrix} 1 & 2 \\ 6 & 5 \end{bmatrix}$$

(10 Marks)

- b. Obtain the topological sort for the graph, Fig. Q4(b) using i) Source Removal method (10 Marks)
 - DFS method.



ig. Q4(b)

Module-3

lolve the Greedy Knapsack problem, Fig, Q5(a) of capacity 5kgs.

(05 Marks)

Fig. Q5(a)

| Items | 1 | 2 | 3 | 4 |
|--------|---|---|---|---|
| Profit | 5 | 9 | 4 | 8 |
| Weight | 1 | 3 | 2 | 2 |

1 of 3

DOWNLOAD THIS FREE AT

www.vturesource.com

18CS42

b. Find the Optimal solution for the Greedy Job sequencing problem given n = 4, profits [10, 30, 60, 40], deadlines [2, 3, 1, 3]. (05 Marks)

c. Apply Prims and Kruskal's algorithm to find the minimal cost spanning tree for the graph given in Fig. Q5(c).

Fig. Q5(c)

7 2 3 6

OR

6 a. A document contains the letters "A" through "E" with frequencies is follows

A: 22, B: 13, C: 18, D: 16, E: 31.

Construct a Huffman Tree and codes and

Encode: CAB, ADD, BAD, ACE Decode: 110011 and 1000110001.

(10 Marks)

b. Apply Heapsort for the list [9, 7, 1, 8, 3, 6, 2, 4, 10, 5] using Bottom up approach.

(10 Marks)

Module-4

Apply Floyd's algorithm to find the all pairs shortest path for the given adjacency matrix.
Fig. Q7(a).

b. Solve the instance of 0/1 Knapsack problem Fig. Q7(b), using Dynamic Programming approach. (10 Marks)

| Item | Weight | Value |
|------|--------|-------|
| 1 | 2 | \$ 12 |
| 2 | 1 | \$10 |
| 3 | 3 | \$ 20 |
| 4 | 2 | \$ 15 |

Capacity W = 5

Fig. Q7(b)

OR

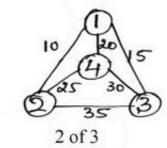
8 a. Construct an Optimal Binary search tree for the set of keys given in Fig. Q8(a). (10 Marks)

| Keys | A | В | С | D |
|-------------|-----|-----|-----|-----|
| Probability | 0.1 | 0.2 | 0.4 | 0.3 |

Fig. Q8(a)

Apply Dynamic programming approach to solve the given Travelling Salesman problem.
 (10 Marks)

Fig. Q8(b)

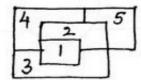


18CS42

Module-5

- With the help of State Space tree, solve the 4 queens problem by using Backtracking approach.
 - b. Color the regions in the Map given in Fig. Q9(b), by applying backtracking graph color algorithm. Color = (R G B & Y). (10 Marks)

Fig. Q9(b)



OR

10 a. Apply LC - Branch and Bound approach to the assignment problem Fig. Q10(a).

Fig. Q10(a) $C = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 9 & 2 & 7 & 8 \\ 6 & 4 & 3 & 7 \\ 5 & 8 & 1 & 8 \\ 7 & 6 & 9 & 4 \end{bmatrix}$ Person a Person b Person c Person d

b. Apply Branch and Bound approach to solve the instance of 0/1 Knapsack problem.

KnapSack Capacity W = 10

| Items | | 2 | 3 | 4 |
|--------|-------|-------|-------|-------|
| Weight | 4 | 7 | 5 | 3 |
| Value | \$ 40 | \$ 42 | \$ 25 | \$ 12 |

Fig. Q10(b)

(10 Marks)