No. of Printed Pages : **2**

Sl. No.

C4-R4 : ADVANCED ALGORITHMS

NOTE :

1. Answer question 1 and any FOUR from questions 2 to 7.

2. Parts of the same question should be answered together and in the same sequence.

Time : 3 Hours **Total Marks : 100**

1. (a) Define time complexity and space complexity of an algorithm.

(b) Write down the recursive algorithm to find the Fibonacci series. Also find out the time complexity of that algorithm.

(c) The fractional knapsack problem can be optimally solved by greedy technique while the 0/1 knapsack problem may or may not be optimally solved with a greedy approach. Why ? Give justification for your answer. .

(d) Prove or give counter example for : "Dynamic Programming avoid calculating the same stuff twice thus better than Backtrack Algorithm".

(e) What is radix sort ? Sort the following elements in increasing order using radix sort technique. 1000, 5000, 2000, 4000, 3000

(f) How to choose an algorithm for solving a single problem, when multiple algorithms for solving a same problem are available ? Which from the below function takes less time to solve the given problem ? Justify your answer.

1. $f(n) = 2f(n-1) + 1$

2. $f(n) = f(n-1) + n$

(g) Define P, NP, NP-Complete and NP-Hard Problems. **(7x4)**

2. (a) What is recurrence ? Solve the following recurrence using Master's Theorem : $T(n) = 4\ T(n/2) + n^3$

(b) Apply bubble sort technique to sort following set of integers : 99, 88, 77, 66, 55, 44

(c) What is order of growth ? Derive best case, average case and worst case time complexity for linear search algorithm. **(8+5+5)**

3. (a) Find the asymptotic bound for the following recurrence using recurrence tree method.

$T\ (n) = T\ (n/2) + T\ (n/4) + T(n/8) + n$

(b) What is searching ? How a binary search algorithm is better than linear search algorithms ? Write down binary search algorithm and derive its time complexity.

(c) Sort the following list in increasing order using quick sort technique. Always consider first element as pivot element.

40, 20, 10, 80, 60, 7, 30, 100 **(6+6+6)**

4. (a) What is approximate algorithm ? Propose an approximation algorithm to solve the problem of Vertex Cover. Through an example, explain how the proposed algorithm approximates the Vertex Cover Problem.

(b) Explain the Strassen's method for multiplying two matrices. Why this approach for solving problem is not preferred for practical applications ? **(10+8)**

5. (a) How many valid match and spurious hits does the Rabin-Karp matcher encounter in the text T = 23590231411526739921 when looking for the pattern P = 31415 ? Take value of modulo q = 13.

(b) What do you mean by Minimum Spanning Tree ? Give Example of it. Also explain Kruskal's algorithm for minimum spanning tree with an appropriate example. **(10+8)**

6. (a) Mention basic four steps of dynamic programming. Find an optimal parenthesization of matrix-chain product whose sequence of dimensions is <5, 10, 3, 5, 6, 2>. Also mention the time complexity and space complexity of matrix chain multiplication algorithm.

(b) Write down the Bellman-Ford algorithm to compute the shortest paths from a single source vertex to all other vertices in a weighted graph.

(c) State True or False with justification : "Two algorithms must be compared only by their worst case behavior." **(8+6+4)**

7. (a) Find out longest common subsequence from the given two sequence of characters : P = <MLNOM> Q = <MNOM>

(b) Show how BFS (Breadth First Search) works on the graph with an example.

(c) Write down an algorithm for naïve string matching algorithm. Derive its time complexity. Also show the comparisons the naïve string matcher makes for the pattern P = 10001 in the text T = 00000100010010. **(6+6+6)**

- o O o -