Ref. No.: Ex/CSE/MCA/T/225A/2018

M.C.A. SECOND YEAR SECOND SEMESTER EXAM 2018

DESIGN AND ANALYSIS OF ALGORITHMS

Time: Three hours **Full Marks: 100**

Answer Question no. 1 and any FOUR from the rest

1. a) Find the recurrence equation for worst case analysis of *Mergesort* algorithm and compute the value of the recurrence equation.

   b) Define (i) the class P (ii) NP-completeness

   c) Compute the value of following recurrence: $T(n) = 3T(n/5) + O(n)$, $T(1) = 1$, $T(0) = 0$. Use Master theorem or other suitable method.

   d) Show that ${}^{n}c_{k}=O(n^{k})$

   e) What is called Memoization? How it is related to dynamic programming? Give a small example.

   4 x 5=20 marks

2. What are characteristics of *divide and conquer* approach to algorithm design? Give the average case analysis of *quicksort* algorithm. Is *quicksort* algorithm really quicker than other sorting algorithms?-explain. 2 + 15 + 3 = 20 marks

3. a) Explain the implementation of *Breadth First Search* algorithm for graph search using the queue data structure. Take an example of a graph and show order of visiting the nodes using the above mentioned BFS algorithm. With proper implementation details, compute running time of the algorithm.

   b) Explain Satisfiablity problem. Is it *NP* complete? - Explain briefly.

   5 + 5 + 5 +5 =20 marks

4. Write *Prim's algorithm* for finding minimum spanning trees (MST). Explain with a suitable example the implementation of this algorithm using linear queue data structure. Compute the running time of the algorithm with the linear queue data structure. Explain how better running time can be achieved using priority queue data structure.

   5 +5 + 5 + 5 = 20 marks

5. What are differences among recursive solution and dynamic programming solution for a problem if both the solutions exist?

   Derive the dynamic programming algorithm for string alignment problem.

   Show with examples the difference between the 0-1 knapsack problem and the fractional knapsack problem. How can the fractional knapsack problem be solved?

   5+10+5=20 marks

6. **A) Use the dynamic programming approach to finding minimum edit distance and the optimal alignment between the strings "VINTNER" and "WRITERS".**

   b) Apply dry-run of 0-1 Knapsack algorithm on the following data and find (i) optimal solution and (ii) the subset of items which gives the optimal solution.

   n = 4 (# of elements)
   W = 10 (max weight)
   Elements (weight, benefit):
   (5, 10), (4, 40), (6, 30), (3, 50)

   10+10=20 marks

7. **A) Why it is important to know that a problem is NP complete or not- Explain**

   c) Is P=NP? Give reasons in support of your answer.

   e) Define "Reductions". What is its importance?

   d) Explain the easiest way of proving a new problem is NP-complete.

   5 x 4 = 20 marks