Ex/IT/T/321/2018

B.E. INFORMATION TECHNOLOGY THIRD YEAR SECOND SEMESTER EXAM 2018

Subject: **DESIGN & ANALYSIS OF ALGORITHM**

Time : Three hours

Full Marks: 100

Different parts of the same question should be answered together.

CO1 [20]	1. Answer any two from {a, b, c}, 2 ×10 =20 a. A postfix expression 'postfix' is with binary operators only. Design a method to generate an equivalent 'infix' expression. Find the complexity of your method. b. Prove that height of the red-black is bounded by O(log n), n is the number of non-NULL nodes. c. i. Prove that for any two function f(n) and g(n), f(n)=Θg(n) if and only if f(n)=O(g(n)) and f(n)=Ω(g(n)). ii. Find the order of growth of $\sum_{i=0}^{n-1}(i^2+1)^2$.
CO2 [30]	2. Attempt any three from {a, b, c, d}, 3 ×10 =30 a. *m* types of coins are available in infinite quantities where the value of each coin is given in the array $C=[c_0,c_1,\ldots,c_{m-1}]$. How many minimum coins needed to make change of S. For example C=[1, 5, 6, 8] and S=11. 11= 5+6, or 11=8+1+1+1, ... i.e., {5, 6}, {5, 1, 1, 1, 1, 1, 1}, {6, 1, 1, 1, 1, 1} or {1, 1, 1, 8} be the solutions. Here, optimal solution is {5,6} since only two coins are needed where in other solutions more coins are required. Write a greedy method to solve this problem. Does your algorithm be able to give optimal solution? b. Write a divide-and-conquer approach to find both minimum and maximum from a data set. Find the complexity of your method. c. Define the longest common subsequence (LCS) problem. Define an optimal solution structure to solve LCS problem using dynamic programming technique. Consider two strings: X="HUMAN" and Y= "CHIMPANZEE" and compute LCS(X, Y). d. Define the string edit distance D(X,Y) between two strings X and Y. Define an optimal solution structure to find D(X, Y) using dynamic programming technique. Consider two strings: X= "PARK" and Y= "SPAKE" and compute D(X, Y).
CO3 [15]	3. Attempt both the questions 5+10=15 a. Prove that the complexity of the comparison based sorting method is bounded below by O(n log n). b. State the order statistics problem. A binary search tree node has two fields: i) integer number and ii) no_nodes. 'no_nodes' represents the number of nodes in the tree rooted at that node. Give a method to locate the k[th] smallest number. OR Let $A_{n\times n}$ be a matrix with numbers, which passes through two steps: i) each row of A sorts in ascending order; ii) each column of A are also sorts in ascending order. After these two steps, each row of A remains in ascending order. Prove it.

P.T.O.

CO4 [15]	4. Answer any one from {a, b} $1 \times 15 = 15$ a. Let S be a set of 2-D points. Define closest pair points on S. Give a method with complexity $O(\|S\| \log \|S\|)$ to find a closest pair from S. Illustrate with an example. b. Let G be a graph. Define shortest path between two vertices 's' and 't' in G. Design an efficient method to find a shortest path between to vertices 's' and 't' of a given graph. Also, find the complexity of your method. Illustrate with an example.
CO5 [20]	5. Answer any two from {a, b, c} $2 \times 10 = 20$ a. Define class P and NP. Define polynomial-time reduction of problem P_1 to problem P_2 ($P_1 \leq_P P_2$). If $P_1 \leq_P P_2$ and $P_2 \in P$, then prove that $P_1 \in P$. b. Assume 3SAT is NP-complete, prove that 'independent set' problem is NP-complete. c. Assume 'vertex-cover' is NP-complete, prove that 'set-cover' problem is NP-complete.

CO1: Recollect notations for algorithm analysis and basic data structures and assess the performance of the associated operations (K3, A2)

CO2: Illustrate and apply different algorithmic paradigms to solve problems and analyze them (K4, A3)

CO3: Analyze, compare and differentiate the behavior of sorting/searching algorithms under different cases and solve the problem. (K4, A3)

CO4: Analyze, compare and distinguish the different graph and geometric algorithms and solve problems (K4, A3)

CO5: Describe and express the concept of NP-completeness and Approximation algorithms. (K2, A2)