2018

COMPUTER SCIENCE AND ENGINEERING

Paper : CSCL-901

(Topics in Algorithms)

Full Marks : 70

The figures in the margin indicate full marks.
Candidates are required to give their answers in their own words as far as practicable.

1. Answer ***any five*** from the following : 2×5

(a) Define approximation algorithm.

(b) Let $\prod_1$ and $\prod_2$ be two problems such that $\prod_1 \leq_P \prod_2$. Suppose that the problem $\prod_2$ can be solved in $O(n^k)$ time and the reduction can be done in $O(n^j)$ time. Show that the problem $\prod_1$ can be solved in $O(n^{jk})$ time.

(c) Compute the prefix function π for the pattern *ababbab* when the alphabet set is $\{a, b\}$.

(d) What is the time complexity of Huffman coding algorithm? Justify.

(e) When will the approximation ratio to solve the vertex cover problem by approximation algorithm exactly equal to 2? Justify your answer.

(f) What can be the maximum length of the augmenting path in a network of bipartite graph of n vertices and m edges? Justify your answer.

(g) Give example of a small network where the Ford-Fulkerson algorithm can take exactly mC iterations, where m is the number of edges and C the maximum capacity of edges.

2. Answer ***any five*** from the following : 4×5

(a) Given an array χ containing n *unique* real numbers, design and analyze an efficient algorithm that finds out a number in χ that is neither minimum nor maximum.

(b) If module q is 11, how many spurious hits does the Rabin-Karp matcher encounter in the text $T = 3141592653589793$, when looking for the pattern $P = 26$?

(c) Given a convex hull H and a point p exterior to the hull H, write an $O(\log n)$ time algorithm to compute the combined hull of H and p.

(d) Show that satisfiability problem (SAT) is polynomial time reducible to Vertex Cover (VC).

(e) Given the following bipartite graph $G = (U \cup V, E)$ with vertices $U = \{a, c, e, g, i\}$ and $V = \{b, d, f, h, j\}$ and $E = \{(a, d),(a, f),(b, c),(b, e),(c, d),(c, f),(c, h),(d, e),(d, g),(e, h),(f, g),(h, i),(i, j)\}$. Find the maximum matching using Hungarian Tree method.

(f) Prove or disprove : Median selection algorithm does not run in linear time when the group size is 3.

Please Turn Over

(g) Suppose I have written a greedy algorithm to solve activity selection problem as follows : select the activity of least duration from among those that are compatible with previously selected activities. Will this algorithm always select maximum number of non-overlapping activities? Justify with an example.

3. Answer ***any four*** from the following :

(a) (i) Write a pseudocode for the naive string matching algorithm to find the first occurrence of a pattern P of length m in a text T of length n. What is the worst-case time complexity of this algorithm?

(ii) Suppose that all characters in the pattern P are different. Show how to accelerate the above naive string matcher to run in time $O(n)$ on an n character text T. 6+4

(b) (i) What is meant by optimal substructure and overlapping subproblems in the context of dynamic programming?

(ii) The subset sum problem is : given a set of n positive integers, $S = \{a_1, a_2, ...a_n\}$, and a positive integer W, is there a subset of S whose elements sum to W? Derive the recurrence to decide if such a subset exists using dynamic programming. Note : $X[i, j]$ denotes true(1) iff there is a subset $\{a_1, a_2, ...a_i\}$ whose elements sum to j. 5+5

(c) Given a set of n convex polygons. Write an efficient algorithm to find the convex hull of the set of n polygons. What is the time complexity of your algorithm? Justify your answer. 6+2+2

(d) (i) Prove or disprove the following statement— "If all capacities in a network are distinct, then there exists a unique flow function that gives the maximum flow."

(ii) In a hospital, there are n doctors. There are two shifts in the hospital : morning and evening. Each doctor has given his/her preference on the day and the shift when he can work. No doctor will be scheduled to work for the entire day, i.e. both morning and evening shifts. Each doctor can be engaged atmost for two days. Each shift must be covered by atmost one doctor. Design an efficient formulation to solve this problem. 4+6

(e) (i) Define P, NP, NP-Complete, NP-Hard complexity classes.

(ii) Consider the following statement. If a problem $\prod \in NP$ and $\prod \leq_P \prod'$ where $\prod'$ is an NP-Complete problem, then $\prod$ is NP-Complete. State whether the above statement is true or false with a brief explanation. You may assume that $P \neq NP$. 6+4

(f) Define Euclidean Travelling Salesman Problem (ETSP). Give a 2-approximation algorithm for ETSP. What is the time complexity of the algorithm? Justify the approximation ratio. 1+3+2+4