MCS-031

MCA (Revised)

Term-End Examination

December, 2008

09189

MCS-031 : DESIGN AND ANALYSIS OF ALGORITHMS

Time : 3 hours *Maximum Marks : 100*

Note : *Question number 1 is compulsory. Attempt any three questions from the rest. All algorithms should be written nearer to C/C++ language. Parts of the same question should be attempted together.*

---

1. (a) (i) What is the difference between a O(big-oh) and O(little-oh) notations. Which of these notations is not asymptotically tight 5

(ii) Arrange the following in the order of worst to best efficiency : 2

$O(n \log n)$, $O(2^n)$, $O(\log n^2)$, $O(n^2)$

(b) Write an algorithm (an informal algorithm is sufficient) to find the i^{th} order statistic, which ensures that the worst case running time to select an element is $O(n)$, i.e., linear. 7

(c) (i) Discuss informally that a recursive solution to matrix-chain multiplication is exponential in nature. 5

(ii) If, both dynamic programming and greedy approaches are applicable to a given problem, then which approach would generally give a faster solution, and why ? 5

(d) What is the benefit of randomization in quicksort algorithm ? 4

(e) What are non-regular languages ? Can we use context free grammar to generate regular as well as non-regular languages ? If yes, then give an example of each. 7

(f) What is the difference between polynomial time and non-polynomial time algorithms ? 5

2. (a) Discuss the role of optimal substructure property in applying a greedy algorithm to solve a given problem. 5

(b) (i) Analyze the worst case running time of merge sort on n numbers. 6

(ii) Given an array of elements : 4

5, 2, 9, 4, 3, 7, 6, 8

If we apply quick sort algorithm on the above array then, what is the minimum and maximum number of times, each element would be compared to others.

(c) Write Dijkstra's Algorithm to find shortest path to a given node in an undirected weighted graph. 5

3. (a) Explain how binary search algorithm performs more efficiently than linear search algorithm. Compare their running times. 5

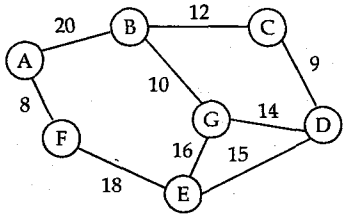
(b) Write a recursive function to find/calculate the sum of all elements in an integer array. 5

(c) Describe, how alpha beta pruning saves time as compared to minimax-procedure in a two player game. You can use an example. 5

(d) What is preconditioning ? Explain how preconditioning is useful in finding a solution to a problem. 5

4. (a) Design a turing machine that accept collection of all strings with an even number of a's and an even number of b's over the alphabet $\Sigma = \{a, b\}$. 7

(b) Using Kruskal's algorithm, construct a minimum spanning tree from the following graph : 8



(c) Write a context free grammar that can generate the language represented by the regular expression (b*a b*a b*)*. 5

5. (a) While applying Depth First Search in a directed graph, how do we differentiate between a cross edge and a forward edge with respect to the discovery time (i.e., the time at which a vertex become known) of its end vertices. 5

(b) (i) What language is represented by the regular expression b*a b*(ab*ab*)*. 3

(ii) Define pumping lemma for regular languages. 2

(c) What is the difference between divide and conquer approach and dynamic programming approach to solve problems. 5

(d) Define an NP-hard problem. Give an example also. 5

- o O o -