Reg. No. : | | | | | | | | | | | | |

Question Paper Code : 13291

M.E./M.Tech. DEGREE EXAMINATION, JANUARY 2015.

First Semester

Computer Science and Engineering

CP 7102 — ADVANCED DATA STRUCTURES AND ALGORITHMS

(Common to M.E. Computer Science and Engineering (with specialization in networks), M.E. Biometrics and Cyber Security, M.E. Multimedia Technology and M.Tech. Information Technology)

(Regulation 2013)

Time : Three hours Maximum : 100 marks

Answer ALL questions.

PART A — (10 × 2 = 20 marks)

1. Differentiate Recursion and Iteration with an example.

2. Give a Recursive algorithm to count the number of nodes in a Binary Tree.

3. What is the running time complexity of 8 Queen's problem? Justify.

4. How map coloring problem is solved using Recursive Backtracking algorithm?

5. What is chromatic number of a graph? Give example.

6. Compare Monte Carlo and Las Vegas algorithms.

7. Define mutual exclusion. Give appropriate example.

8. Define the structure of petri net.

9. What is lazy synchronization?

10. What is the role of locking in synchronization?

PART B — (5 × 16 = 80 marks)

11. (a) (i) Write an algorithm to search a key k in a $N * N$ Matrix in which each row is sorted in ascending order. What is the time complexity of your algorithm? (8)

(ii) Give a recursive algorithm to find the k^{th} smallest element of a set S. Write the recurrence equation and perform asymptotic analysis for worst case. (8)

Or

(b) (i) Write an algorithm to find the next least node in a binary search tree given a node and assess the time complexity of the algorithm. (8)

(ii) Given a set of integers with repeated occurrences of elements. Write an algorithm to print the power set of S ensuring that the repeated elements are printed only once. For example, $S = \{1,2,2\}$. For the above S the power set will be {NULL, {1}, {2}, {2}, {1,2}, {1,2}, {2,2}, {1,2,2}}. The final output should print {NULL, {1}, {2}, {1,2}, {2,2}, {1,2,2}}. (8)

12. (a) (i) Give the computational complexity of single source shortest path algorithm for the following graph representations

(1) Adjacency matrix representation.

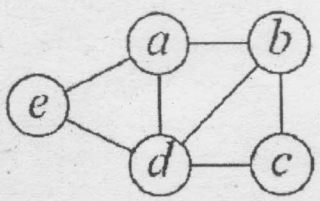
(2) Adjacency list representation. (8)

(ii) Write an algorithm to find sum of any path from root to leaf such that the sum of all nodes along the path is maximum compared to all other path. Analyze the algorithm. (8)

Or

(b) (i) Solve Travelling Salesman Problem with Hill Climbing Algorithm. Describe the solution. (8)

(ii) What is acyclic path? Write an algorithm to print all acyclic paths in a given graph from a node a. (8)

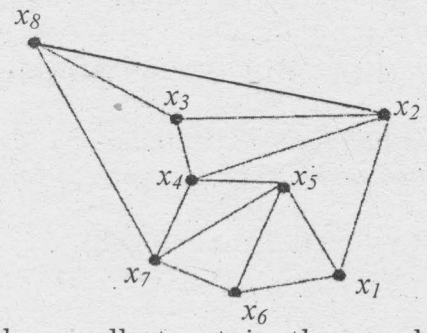


13. (a) (i) Construct an efficient algorithm to calculate the length of the longest common substring in the given two strings. (8)

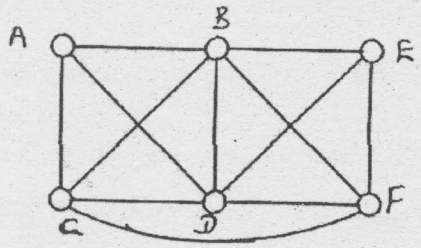
(ii) Consider a bipartite graph G = (V, E) with bipartition (A, B): V = $A \cup B$. Assume that, for some vertex sets $A_1 \subseteq A$ and $B_1 \subseteq B$, there exists a matching M_A covering all vertices in A_1 and a matching M_B covering all vertices in B_1. Prove that there always exists a matching covering all vertices in $A_1 \cup B_1$. (8)

Or

(b) (i) What is the three-color problem? Check whether the given graph is 3-colorable. (8)



(ii) What is the smallest cut in the graph? Write an algorithm to compute the minimal cut in the graph shown below and show the output. (8)



14. (a) (i) Prove an upper bound on the time required for a particular process to reach its critical region from the time when it enters in the Lamport's bakery algorithm. (8)

(ii) What is the role of semaphores in solving producer consumer problem? Explain. (8)

Or

(b) (i) Describe the steps in solving mutual exclusion problem among n processes namely $p_1, p_2, \ldots, p_n$ Write an algorithm for process p_i. (8)

(ii) Explain the strategies for developing Parallel Applications. (8)

15. (a) (i) Describe the data structures and code used for concurrent queues. (8)

(ii) Compare coarse-grained and fine-grained synchronization. (8)

Or

(b) (i) State ABA problem. Explain the solution with an example. (8)

(ii) Design an unbounded lock-based stack on a linked list. (8)
