2011

MCA

2nd Semester Examination

DESIGN & ANALYSIS OF ALGORITHMS

PAPER—CS/MCA/201

Full Marks : 70

Time : 3 Hours

The figures in the margin indicate full marks.

Candidates are required to give their answers in their own words as far as practicable.

Illustrate the answers wherever necessary.

Answer any *five* questions.

1. (a) What do you mean by performance measurement and performance analysis. 5

(b) Devise an algorithm that searches an unsorted array a[1 : n] for the element x. If x occurs, then return a position in the array; else return zero. Also, find the worst case complexity of the algorithm. 6+3

2. (a) What is a N-Queen's problem? How do we solve it? 3+3

(b) Write an algorithm to solve the above problem, give comments wherever necessary. 5

(c) State the features of an algorithm. 3

3. (a) Consider the recurrence $T(n) = 7T(n/2) + 18n^2$, $n \geq 2$ and a power of 2. Find the growth of function T(n) using. Divide and conquer technique. 4

(b) How do you analyze an algorithm? 5

(c) Define Asymptotic Notation? What do you mean omega notation. 2+3

4. (a) How do we approach divide and conquer problem. 4

(b) Write an algorithm for quicksort, explain briefly, wherever necessary. Also, find the time complexity in worst case. 6+4

5. (a) If $f(n) = a_m n^m + ... + a_1 n + a_0$ and $a_m > 0$, then prove that $f(n) = \Omega(n^m)$.

5

(b) What is a Greedy Method Technique. Write an algorithm to solve Fractional Knapsack problem.

3+4

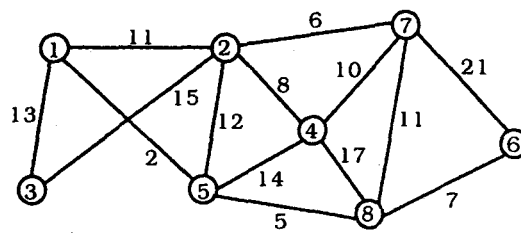
(c) Write an algorithm to solve Towers of Hanoi problem.

2

6. (a) What are the features that emphasizes a problems to be solved by dynamic programming? State briefly.

5

(b) Write a prim's Algorithm and find MST for the below graph using prim's Algorithm. 4+5



7. (a) Given a chain of matrices $(A_1, A_2, \ldots\ldots A_n)$. A_i has dimension $p_{i-1} \times p_i$, for i = 1, 2, ... n. The problem is to fully parenthesize the product $A_1 A_2 \ldots\ldots A_n$ in a way that minimizes the no. of scalar multiplication. Solve the above problem using suitable algorithm. 10

(b) Analyse Merge sort to find the time complexity. 4