Exam No. _____

GANPAT UNIVERSITY

B. Tech. Semester VI (Computer Engineering / Information Technology)

Regular Examination April - June 2015

2CE604 / 2IT604 : Design and Analysis of Algorithms

Time: 3 Hours **Total Marks: 70**

Instruction: 1. Each section should be written in a separate answer book.
2. Be precise and to the point in your answer.

Section - I

Q – 1 (a) Explain big-omega notation with example. **(4)**

(b) Find out time complexity of following algorithm: **(4)**

```
Matrix mul()
{
    for i=1 to n
        for j=1 to n
         c[i][j]=0
           for k=1 to n
              c[i][j] = c[i][j]+a[i][k]*b[i][k]
}
```

(c) Solve the recurrence using intelligent guesswork method: **(4)**
$T(n) = 2T(n-1) + 1$ using $T(0)=0$

OR

Q – 1 (a) Explain space complexity. Discuss how recursion affects the space complexity. **(4)**

(b) Prove whether following statements are true or false: **(4)**
1. $f(n) = 9n^2 + 2n + 5 = O(n^3)$ 2. $f(n) = 3n^3 + 7n^2 + 5 \neq \Omega(n^4)$

(c) Solve the recurrence: **(4)**
$T(n) = T(n/3) + T(2n/3) + n$

Q – 2 (a) Solve the recurrence using change of variable method: **(6)**

$$T(n) = \begin{cases} 1 & \text{, if } n = 1 \\ 3T(n/2) + n & \text{, when n is power of 2, } n > 1 \end{cases}$$

(b) Compare normal exponentiation with fast exponentiation. Write algorithm of fast exponentiation. **(5)**

OR

Q – 2 (a) Solve the recurrence: **(6)**

$$T(n) = \begin{cases} 1 & \text{, if } n = 0 \\ 2 & \text{, if } n = 1 \\ 4T(n-1) - 3T(n-2) + 1 & \text{, otherwise} \end{cases}$$

(b) Explain Counting inversion using Divide and Conquer Technique with example. **(5)**

Q – 3 (a) Prove that $\max(f(n), g(n)) = \Theta (f(n) + g(n))$ **(4)**

(b) Solve the recurrence: **(4)**
$T(n) = 4T(n/2) + \log n$

(c) Explain time complexity analysis of quick sort for best, average and worst case. Which sorting algorithm will give best time complexity in case of elements are almost sorted? **(4)**

Section - II

Q – 4 (a) Using dynamic programming, find the a) optimal parenthesization and b) minimum number of multiplications, for the given matrices in matrix chain multiplication problem **(6)**

A: 3 x 2 **B:** 2 x 5 **C:** 5 x 1 **D:** 1 x 10

(b) Using Dynamic Programming, solve the following integer knapsack instance: **(6)**
Capacity = 6, [w1, w2, w3] = [3, 5, 4] and [p1, p2, p3] = [8, 18, 14] where wi & pi are the weight & profit of i^{th} object respectively.

OR

Q – 4 (a) Given unlimited coins of denominations **4, 3** and **1** and the amount to pay is **6** for the making change problem. Find optimal solution using **(6)**

a) Dynamic programming method b) Greedy Method

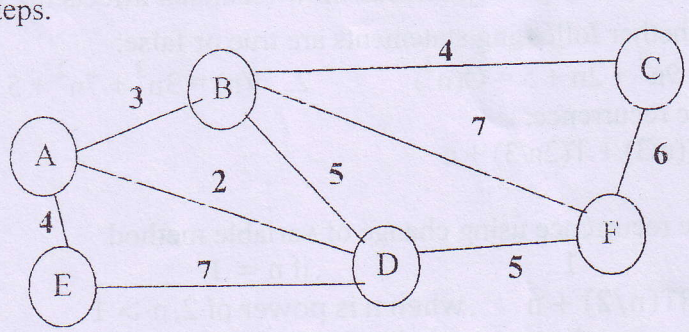
Also state which technique is better for making change problem and why?

(b) Using dynamic programming approach, find the longest common subsequence from the given two sequences: Seq1="LATE" & Seq2="ASTRE" **(6)**

Q – 5 (a) Assuming that in the job scheduling problem each job takes one unit of time. Find the optimal job schedule for the given list of jobs with deadlines & profits mentioned below. Also find the total profit. **(6)**

Job_ID	J1	J2	J3	J4	J5	J6
Deadline	2	2	5	1	3	3
Profit	5	15	12	20	25	10

(b) Using Prims' algorithm, find the minimum cost spanning tree for the given graph. Also show intermediate steps. **(5)**



OR

Q – 5 (a) What is dynamic programming approach? What type of problems can be solved using Dynamic programming Approach? Differentiate between Divide & Conquer and Dynamic programming approach with suitable examples. **(6)**

(b) Write the Kruskal's algorithm for finding minimum cost spanning tree. **(5)**

Q – 6 (a) Explain Breadth First Search with Example. **(6)**

(b) Discuss the concept of N-Queens problem. Give one solution of 8-Queens Problem using Backtracking Method. **(6)**

END OF PAPER