Shri G.S. Institute of Technology & Science

Computer Engineering Department

CO4407: Design and Analysis of Algorithms (Test - III) Oct-2014

Time: 60 Mins] **[Max. Marks 25**

Q. 1	Write a dynamic programming algorithm to determine the best factorization of a for the matrix chain $A_1A_2A_3A_4A_5$ with dimensions p = <3,10, 50, 100, 10, 25>. Construct a table and determine the factorization and minimum number of multiplications.	[6]
Q. 2	Design and analyze an algorithm to find the length of the longest sub-sequence of a given string that is also a palindrome. For example, the longest palindrome sub-sequence of MAHDYNAMICPROGRAMZLETMESHOWYOUTHEM is MHYMRORMYHM, so given that string as input, your algorithm should output the number 11.	[6]
Q. 3	A shuffle of two strings X and Y is formed by interspersing the characters into a new string, keeping the characters of X and Y in the same order. For example, 'bananaananas' is a shuffle of 'banana' and 'ananas' in several different ways. bananaananas bananaananas bananaananas Given three strings A[1..m], B[1..n], and C[1..m+n], design and analyze an algorithm to determine whether C is a shuffle of Aand B.	[6]
Q. 4	Suppose that you have n unit length jobs to schedule on m processors. Typically n >> m. The ith job has a list L^i of the processors on which it can be run. Each job i is to assigned to exactly one processor on its list L^i. Multiple jobs can be assigned to the same processor, however, each processor j has a job limit P^j which is the maximum number of jobs that can be assigned to processor j. Design an efficient algorithm which either finds a legal assignment of all jobs or determines that no such assignment exists. Be sure to justify the correctness of your solution algorithm. Also give the running time of your solution as a function of n and m.	[6]
Q. 5	Prove that the finding the n the Fibonacci number is not a NP problem.	[1]

Shri G.S. Institute of Technology & Science

Computer Engineering Department

CO4407: Design and Analysis of Algorithms (Test - II) July-2014

Time: 50 Mins] **[Max. Marks 25**

Note: Open book test [you can carry any printed materials books/notes/printouts in the exam hall].

Q 1. Find the solution of the given below recurrence relation in the terms of n and m. **[5]**

$T(n) = 8T(n/2) + \theta(1)$ if $n^2 > m$

$= m$ if $n^2 \leq m$

Q 2. Rank the following functions by order of growth, for the large value of n: **[5]**

$(\log n)^{\log n}$, $(1/n)^n$, $n!$, $(\log n)!$, 2^{n-1} , e^n , $\sqrt{n}$, $n \log n$, $\log \log n$, n^2

Q 3. Design an algorithm to find the n^{th} Fibonacci number in the (log n) time. **[5]**

Q 4. Design an algorithm to find the 2^{nd} largest element in the list of n elements in (n+logn-2) time. **[5]**

Q 5. Given a matrix (NxN) in which each row and each column are sorted in increasing order, Design and analysis (by using recurrence relation) a divide and conquer algorithm to find an element in it. **[5]**