Analysis and Design of Algorithms

MAJOR EXAMINATION - (Sem 1501)

Time: 2 hours **November 24, 2015** **Marks: 38**

Q1. We are given n jobs that each take one unit of processing time. All jobs are available at time 0, and job j has a profit of cj and a deadline dj . The profit for job j will only be earned if the job completes by time dj.

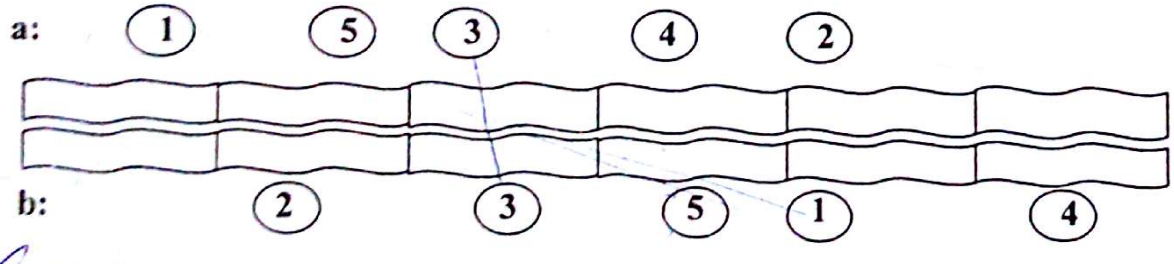
a) Prove that if a subset of the jobs can be completed on time, then they can also be completed on time if they are scheduled in the order of their deadlines.

b) Let E = {1, . . , n} and let I = {J ⊆ E : J can be completed on time }. Prove that M = (E, I) is a Matroid.

c) If you are presented with 10 jobs to be done at time t = 0, with the c and d vectors are as follows :-
c = {3,2,6,7,4,5,6,1,9,10} and d = { 21,5,3,6,9,10,11,15,8,20}. Find the optimal ordering of the jobs in order to earn the maximum profit. Prove that the algorithm suggested by you works in all the cases.

[2 + 3 + 2 = 7]

Q2. Consider a 2-D map with a horizontal river passing through its center. There are n cities on the southern bank with x-coordinates a(1) ... a(n) and n cities on the northern bank with x-coordinates b(1) ... b(n) . Both arrays a and b are permutation of numbers from 1 to n . For illustration consider the diagram below with n = 5. Government wants to connect as many north-south pairs of cities as possible with bridges such that no two bridges cross. When connecting cities, the policy is to connect city *i* on the northern bank to city *i* on the southern bank.



a) Design a polynomial time algorithm to maximize the no. of bridges made.

b) Provide its proof of correctness.

c) Analyze the time and space complexities of your algorithm.

[4 + 2 + 2 = 8]

Q3. Given a matrix of letters write an algorithm to find whether a given input word is present in the matrix. A letter sequence $w_1, w_2, \ldots w_k$ is said to be present in the matrix if w_{i+1} is one of the neighbours w_i for i = 1 to k-1. All the eight movements (i.e. left, right, up, down, diagonals) are allowed to check neighbourhood. Also, find the number of sequences that can form the given word.

a) Write an algorithm to solve the problem for a given input matrix and an input string *s*, and print for all the occurrences the sequence of matrix cells forming the input string *s*.

b) What will your output be for the given matrix, and the input word 'delhi' for all possible occurrences? Explain with appropriate diagrams.

d	e	j	h	y	d
h	d	l	j	m	e
i	k	e	h	k	l
j	h	l	l	y	n
d	f	b	l	d	h
e	l	i	h	e	i

[4 + 3 = 7]

Q4. Consider the two problems SET-PARTITION and SUBSET-SUM as defined below:

SET-PARTITION: Given a set of integers A does there exist a partition S and S^c of A s.t

$$\sum_{x \in S} x = \sum_{y \in S^c} y$$

SUBSET-SUM: Given a set of integers X and a target number N, does there exist a subset S of X such that $\sum_{x \in S} x = N$.

Given that SUBSET-SUM is NP-Complete - show that SET-PARTTION is NP-Complete.

[6]

Q5. a) Give two different definitions for the class NP. Are they equivalent? Justify your answer.

b) Justify that 0/1Knapsack belongs to the problem class NP by designing an appropriate non-deterministic algorithm that runs in polynomial time..

[5 + 5 = 10]