CSE 431/531 Homework 3

Your Name:________________ Your University ID:________________

Problems	1	2	3	Total
Max. Score	12	14	14	40
Your Score				

Problem 1 (12 points). Let $G = (V, E)$ be a directed acyclic graph with weight function: $w : E \to \mathbb{R}$. You may assume that $V = \{1, 2, \cdots, n\}$ and for all edges $(i, j) \in E$ we have $i < j$. The graph is given using the adjacency-list representation. Design an $O(n + m)$-time algorithm to check if the shortest path from 1 to n in G is unique or not.

Problem 2 (14 points). We call a sequence $X[1..m]$ of numbers oscillating if $X[i] < X[i + 1]$ for all even $i \leq m - 1$, and $X[i] > X[i+1]$ for all odd $i \leq m - 1$. For example, $(5, 3, 9, 7, 8, 6, 12, 11)$ is an oscillating sequence since $5 > 3 < 9 > 7 < 8 > 6 < 12 > 11$.

Describe an $O(n^2)$-time dynamic programming algorithm to compute the length of the longest oscillating subsequence of an array $A[1..n]$ of integers. For example, if the input sequence is $(3, 5, 1, 2, 9, 10, 8, 6, 7)$, then your output should be 5, since $(5, 1, 10, 6, 7)$ is a longest oscillating subsequence of the input sequence.

Problem 3 (14 points). We can use dynamic programming to solve the shortest-path problem on directed acyclic graphs (DAGs). With an obvious modification, the algorithm works for the *longest-path* problem on DAGs.

This problem asks you to reduce the weighted interval scheduling problem to the longest-path problem on DAGs. Given a weighted interval scheduling problem $(n, (s_1, s_2, \cdots, s_n), (f_1, f_2, \cdots, f_n))$, you need to show how to construct a DAG $G = (V, E)$ with edge weights $w : E \to \mathbb{R}$ and specify two vertices $s, t \in V$, such that solving the weighted interval scheduling instance is equivalent to computing the longest path from s to t in G. The graph G you constructed must have $|V| = O(n)$.