Spring 2011 **CMSC 451: Homework 1** Clyde Kruskal

Due at the start of class Thursday, September 22, 2011.

Problem 1. Let $G = (V, E)$ be a directed graph. The *reversal* of G is a graph $G^R = (V, E^R)$ where the directions of the edges have been reversed (i.e. $E^R = \{(i, j) | (j, i) \in E\}$).

(a) Assuming that G is represented by an adjacency matrix $A[1..n, 1..n]$, give an $O(n^2)$-time algorithm to compute the adjacency matrix A^R for G^R.

(b) Assuming that G is represented by an adjacency list Adj$[1..n]$, give an $O(n+e)$-time algorithm to compute an adjacency list representation of G^R.

Problem 2. Consider a rooted DAG (directed, acyclic graph with a vertex – the root – that has a path to all other vertices). Give a linear time $(O(|V| + |E|))$ algorithm to find the length of the longest simple path from the root to each of the other vertices. (The length of a path is the number of edges on it.)

Problem 3. Give an algorithm to find all possible topological sorts of a directed graph. What can you say about its running time?

Problem 4. Consider the incorrect algorithm that tries to find strongly connected components by processing the nodes in the second pass in order of finish time.

(a) Briefly give our intuition of why this is a good idea.

(b) Show that the algorithm is incorrect (by giving a counterexample).

(c) Briefly explain why our intuition was incorrect.

Problem 5. Do Exercise 6 on page 108 of Kleinberg and Tardos.

Problem 6. A *bridge* in a connected, undirected graph $G = (V, E)$ is an edge whose removal diconnects G. Give an efficient algorithm to find all of the bridges in G. Write the pseudo-code. Try to make your algorithm clean and elegant. Justify the correctness of your algorithm.