CPS 130: Homework 7

More Graphs: Spanning Trees, Shortest Paths
(CLRS: Chapter 23.2, 24.3, 25.2)

Date on which distributed: Thursday, November 01, 2001
Date on which due: Tuesday, November 13, 2001

Note: Zero credit will be given for homeworks submitted late.

1. [Exc 23.2-2 CLRS]
Suppose that the graph \( G = (V, E) \) is represented as an adjacency matrix. Give a simple implementation of Prim’s algorithm for this case that runs in \( O(V^2) \) time.

2. [Exc 23.2-4 CLRS]
We know that Kruskal’s algorithm runs in \( O(E \lg V) \) time on a graph \( G = (V, E) \). Suppose that all edge weights in \( G \) are integers in the range from 1 to \( |V| \). How fast can you make Kruskal’s algorithm on \( G \)? What if the edge weights are integers in the range from 1 to \( W \) for some constant \( W \)? (You can assume \( W \) is an integer too.)

3. [Exc 24.3-4 CLRS]
We are given a directed graph \( G = (V, E) \) on which each edge \((u, v) \in E\) has an associated value \( r(u, v) \) which is a real number in the range \( 0 \leq r(u, v) \leq 1 \) that represents the reliability of a communication channel from vertex \( u \) to vertex \( v \). We interpret \( r(u, v) \) as the probability that the channel from \( u \) to \( v \) will not fail, and we assume that these probabilities are independent. Give an efficient algorithm to find the most reliable path between two given vertices. (Hint: The probability of the success of any path from \( u \) to \( v \) is the product of the probabilities of success of the individual edges that make up this path. You may want to try to convert the problem to a Dijkstra-like situation. If this is your approach, remember that Dijkstra’s algorithm finds the shortest path between any two given vertices, and that weights of paths are computed as sums, not products of weights of edges along them.)

4. [Exc 25.2-6 CLRS]
How can the output of the Floyd-Warshall algorithm be used to detect the presence of a negative-weight cycle?