ITK 279
Fall, 2012
Califf

Programming Assignment 4:
Working with Graphs

Due date: Friday, November 16, 2012 at 11 pm

Description:
You are to develop a C++ class that represents a weighted, directed graph. You may represent your graph using either of the two standard graph representations discussed in class. Your class must **clearly** use either an adjacency matrix or an adjacency list representation. Your class must provide at least the following public methods:

- ReadGraph will accept the name of a file and read in a new graph from that file.
- TopologicalSort will print to standard output a topological sort of the graph or an error message indicating that a topological sort of the graph is not possible.
- ShortestPath will accept a node name and print to standard output the cost and the actual path to each other vertex that can be reached (one path and cost per line). Must use Dijkstra's algorithm. Do not print anything for vertices that cannot be reached.
- MinimumSpanningTree will print to standard output the weight of the minimum spanning tree and a list of the edges in the tree along with the weight of each (one edge per line). For the purposes of this process, treat the graph as undirected (that is – ignore the direction of edges). Your method may have the graph being connected as a precondition (but your comments must make it clear if you do).

If you are allocating memory dynamically, your class MUST handle it correctly.

You must also write a program that provides a menu to a user allowing the user to input graphs and perform the various graph operations. Your menu should be clear and easy to use, conforming to good user interface standards. Your program must allow for the reading of multiple graphs in a single run of the program.

Input file format:
The input file will consist of 4 parts:

1. The number of vertices
2. The vertex names, one per line – these will be character strings which will not include spaces
3. The number of edges
4. Edge data with one edge per line – first vertex, second vertex, weight

There are sample input files in /home/mecalif/public/itk279/Program4.

Submit your code to ReggieNet.