

# Computer Science 2300: Lab 3

Due: February 18, 2009

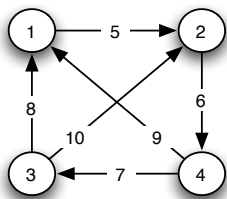
For this lab you will be implementing a very specific version of Dijkstra's algorithm. Make sure you read all instructions carefully.

## 1 Implementing Dijkstra's algorithm

Write a program that implements Dijkstra's algorithm. **Make sure you use a binary heap for the priority queue.** The program takes as input a file (`data.txt`), in which each line either represents a vertex or an edge. For the edge lines, the first integer on that line representing the starting vertex, the second the ending vertex, and the third the weight of the edge. Use this file to construct, line by line, the graph upon which Dijkstra's algorithm will be run (do NOT hardcode this graph!). For example, consider the following text file:

```
<sample.txt>
V 1
V 2
V 3
V 4
E 1 2 5
E 2 4 6
E 4 3 7
E 3 1 8
E 4 1 9
E 3 2 10
```

The graph corresponding to this text file is the following:



For your convenience, a graph class has been provided for you (`lab3-files.tar.gz`). Edit the `dijkstra.cpp` found in this file to implement your program. You may additionally modify the graph class in whatever way you wish in order to complete this lab (with the exception of hardcoding solutions). **Note: while all basic functions for creating a directed graph have been provided for you, you will need to implement all additional functions needed for Dijkstra's algorithm.** It is therefore recommended that you take the time to familiarize yourself with the graph class.

## 2 Compiling and Running Your Code

in addition to `data.txt`, a basic Makefile is included in `lab3-files.tar.gz`. To compile your program, simply type in the command

```
make
```

The makefile will then compile your code with the graph class, and create an executable, `dijkstra`. For your convenience, `dijkstra.cpp` originally contains some starting functions to help you get familiarized with the provided graph class.

To receive full credit, run Dijkstra's algorithm on the graph, starting from node 1. Show your output to the TA to receive full credit.