CQUPT – University at Albany
Computer Science – International College

ICSI 403 --- Design and Analysis of Algorithms

Programming Assignment 4 --- Spring 2025

Assigned: Tuesday, May 6th, 2025.

Due date: Your co-instructor will inform you the date this exercise is to be submitted.

Purpose of the Project

The primary goal of this exercise is to develop skills to work with graphs. A secondary goal is to apply both *Bellman-Ford* and *Dijkstra's* algorithms to compute the shortest path for a problem represented as a graph.

Description

You will be given data in the form of nodes and edges. The first line of the data file is the number of nodes. Nodes are identified by integers and will go from 0 to the number of nodes – 1. Next is the start node. Following the start node is a list of edges and weights. Each line will contain a source node, destination node, and a weight (real number). The end of data is marked by the end of file.

Details

The input data may represent cities. You are to use both the Bellman Ford and the Dijkstra's algorithm to determine the shortest path for graphs that are described by the format defined in this exercise. Your solution must work for graphs that have at most 100 nodes. You are to use the *Priority Queue* you have designed for your Project 2. You must add a header file *mypq.h* to your solution. This file will contain the C++ function declarations and any macro definitions that are associated with your priority queue implementation and will be shared by other source files, used for your solution, that require a priority queue. Your solution will have the *#include "mypq.h"* statement as part of the header declaration of the components that manipulate the priority queue.

Input Data Format Example

The data in your testing files must follow the format defined here. The example that follows illustrates the format to be used.

```
6
1
0  1  12.0
0  2  8.5
0  3  5.5
2  4  4.0
4  5  10.0
3  5  15.0
```

Output Format

Your solution must report the shortest-length path for all nodes starting from the defined origin as determined by the associated input file. You are to place your solution in a text file. The format to

be followed is described below for a graph with 5 vertices <*S, T, X, Y, Z*>, and *S* as the start node. Note: Both the path and the integers used for the example below are just for the description of the format to be used.

The shortest-length path is:

S = 0; path = < >
T = 4; path = < S, Y, T>
X = 9; path = <S, Y>
Y = 5; path = <S, Y, Z>
Z = 7; path = <S, Y, T, X>

How to Submit your Work

- Your solutions must be submitted to the address provided by your co-instructor.
- Your files for the *Project4* must include the source code for all modules used for the exercise as well as an executable file to allow your solution to be tested. You must include all your source files in a Compressed (zipped) folder (.zip). Your (.zip) folder as well as the subject of your message must follow the format: *Project 4 Your Name*. Marks will be deducted if you do not follow this requirement.

Documentation

Your program must be developed using the C++ programing language. It should be layered, modularized, and well commented on. The following is a tentative marking scheme and what it is expected to be submitted.

1. *External Documentation* including as many pages as necessary to fulfill the requirements listed below.
    a. Title page
    b. A table of contents
    c. [5%] System documentation
        i. A high-level data flow diagram for the system
        ii. A list of routines and their brief descriptions
        iii. Implementation details including information such as data structures and algorithms used.
    d. [25%] Test documentation
        i. [5%] How you tested your program
        ii. [20%] Testing outputs. At least two additional different testing files are expected to be used. Such files must have 10 nodes (vertices) each.

    e. [5%] User documentation
        i. How to run your program
        ii. Describe parameter (if any)

2. Source Code
    a. [60%] Correctness (10% for each correct answer for each of the 3 testing files.)

b. [5%] Programming style
    i. Layering
    ii. Readability
    iii. Comments
    iv. Efficiency