CSCI 2270 – Data Structures

Instructors: Christopher Godley, Maciej Zagrodzki

Assignment 9 - Graph

OBJECTIVES

1. **Applications of DF Traversal**
2. **Dijkstra's algorithm**

Overview

In this assignment, you will apply DF traversal for finding connected cities in a graph and find the shortest path using Dijkstra's algorithm.

Graph Class

Your code should implement graph traversal for cities. A header file that lays out this graph can be found in **Graph.hpp** on Moodle. *As usual, do not modify the header file. You may implement helper functions in your .cpp file if you want as long as you don't add those functions to the Graph class.*

Your graph will utilize the following struct:

```
struct vertex;

struct adjVertex{
    vertex *v;
};

struct vertex{
    vertex() {
        this->visited = false;
        this->distance = 0;
        this->pred = NULL;
    }
    string name;
    bool visited;
    int distance;
    vertex *pred;
    vector<adjVertex> adj;
};
```

void addVertex(string name);

- ➔ Use from the previous assignment.

void addEdge(string v1, string v2, int num);

- ➔ Make a connection between v1 and v2 (same as last assignment). **Important** point here is to **add weights to the edges**. Each edge will have a corresponding weight attached to it. E.g. addEdge("Aurora", "Bloomington",5);

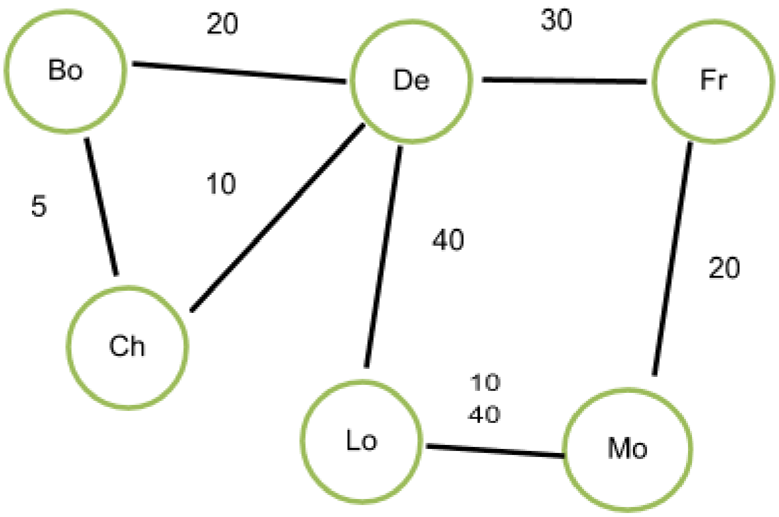
void depthFirstTraversal(string sourceVertex):

- ➔ Use Depth first traversal (recursive) from sourceVertex to traverse the graph. Print the city name and corresponding distance from the sourceVertex. Format for printing:

```
// for printing the path found through DF Traversal (each node)
cout<< n->name << " --> ";
// print “Done” at the end when all the cities have been visited.
cout << "Done";
```

vertex* DijkstraAlgorithm(string start, string end):

- ➔ Use Dijkstra's algorithm to find the shortest path in the graph from the start city to the end city.



For example: Dijkstra's Shortest distance from Boulder (Bo) to Denver (De): 15

void shortestpath(string s1, string s2):

- ➔ You should print the shortest path found though the Dijkstra's algorithm.

```
// for printing the shortest path
cout << path[i]->name << " ";
```

Other cout statements:

cout<<"Depth First Traversal " <<endl;
cout<<endl;
cout<<"Dikstra's Shortest distance from Aurora to Fruita: "<<vertex->distance<<endl;
cout<<"Dikstra's Shortest path from Aurora to Fruita: "<<endl;

Please note that once you are done with your assignment on code runner you need to click on 'finish attempt' and the 'submit all and finish'. If you don't do this, you will not get graded."