IIT Ropar
CSL201 Data Structures
Semester 1, AY 2016/17

Lab Assignment 5 - 30 marks Due on 15th November, 11:59 PM

Objective

Using undirected graphs data structures and applying graph algorithms.

Instructions

1. DO NOT use STL data structures. You have to implement all data structures on your own.
2. You are to use C++ programming language to complete the assignment.
3. Provide a Makefile to compile your final code, keep class definitions in .cpp file and class declarations in .h file.
4. This is an individual assignment. You can have high level discussions with other students, though.
5. Include a “Readme.txt” file on how to compile and run the code. Your program should run on institute ubuntu machines.
6. Upload your submission to moodle by the due date and time. After due date, your submission will be evaluated with 10% penalty per day for next two days. After that, your submission will not be evaluated.

Question [30 marks]

In this assignment you are given a list of countries and its neighbors. The data file (countries.dat) is in the following format:

```
Country Name >> Neighbor Country Name#1[: XX km]; Neighbor Country Name#2[: XX km];
```

Here [:] means it is optional. In most cases, the country has a border, but there are a few cases that do not. Note that every line ends with a semicolon.

Example

--

```
Adélie Land (France) >> Australian Antarctic Territory (Australia);
American Samoa (United States) >> ;
Ecuador >> Colombia: 590 km; Peru: 1,420 km;
```

--

In the first example, Adélie Land (France) has only one (1) neighbor, which is Australian Territory (Australia). However, no border length is given this case. For such examples, you can assume that the border length is 0, however, do not forget to add the neighboring country to the graph.

In the second example, American Samoa (United States) has no neighbors, however, this country should still be added to the graph. The examples of Adélie Land and American Samoa are territories of other countries (in this case France and US respectively), however, in this assignment we will treat them as separate countries.

The third example Ecuador >> Colombia: 590 km; Peru: 1,420 km; Ecuador, has two (2) neighbors. The border between Ecuador and Colombia is 590km, the border between Ecuador and Peru is 1420 km.

Your assignment will be to write a menu-driven console interface that allows the user to list the countries and check if paths between two countries exist (and if so, what is the shortest path in terms of # of countries visited [note that border length does not matter in the shortest path computation]). Additional functionalities must also be provided; see details in the following for additional functionalities.

Example of menu

========== Assignment 5 ==========
1. List all countries and bordering countries
2. Find shortest path
3. List all countries with X neighbors
4. Find countries with borders larger than X km
5. Exit
>

Details

1. List all countries and bordering countries

List all countries by integer ID [0 ~ N-1] and its neighboring countries.

2. Find shortest path

Find shortest path between two countries in terms of number of countries visited (you do not need to factor in the countries border in the shortest path computation). If a path does not exist, output that there is no path.

3. List all countries with X neighbors

List all countries with N neighbors, where the user provides the size of N.

4. Find countries with borders larger than X km

Find all countries with borders larger than X (where the user inputs X). Print the country and its neighbors.

Required data structure

Obviously, you have to implement a graph data structure to complete this assignment.