ITU Computer Engineering Department

BLG 223E Data Structures, FALL 2021

Homework #2

Due Date: 30 December 2021, 13:30

PROBLEM DEFINITION

Two friends are living in different cities. They both have a car and want to meet at a point such that total duration of their journey is minimized.

They both have a highway duration map that shows which cities may be reached from which cities in what a duration. For each friend, the highway duration map is given in a file with the following format:

```
SOURCE_CITY           DESTINATION_CITY       DURATION
………………..           ……………………….        …………….
………………..           ……………………….        …………….
………………..           ……………………….        …………….
SOURCE_CITY           DESTINATION_CITY       DURATION
```

RUNNING THE PROGRAM

The names of the highway duration map files are supplied via command line as follows:

```
./blg223e_hw2     name_of_file_for_friend_1     name_of_file_for_friend_2
```

For example, if the **name_of_file_for_friend_1** is **map1_friend1.txt** and **name_of_file_for_friend_2** is **map1_friend2.txt**, then the program will be run with the following command

```
./blg223e_hw2     map1_friend1.txt     map1_friend2.txt
```

and if the **name_of_file_for_friend_1** is **map2_friend1.txt** and **name_of_file_for_friend_2** is **map2_friend2.txt**, then the program will be run with the following command

```
./blg223e_hw2     map2_friend1.txt     map2_friend2.txt
```

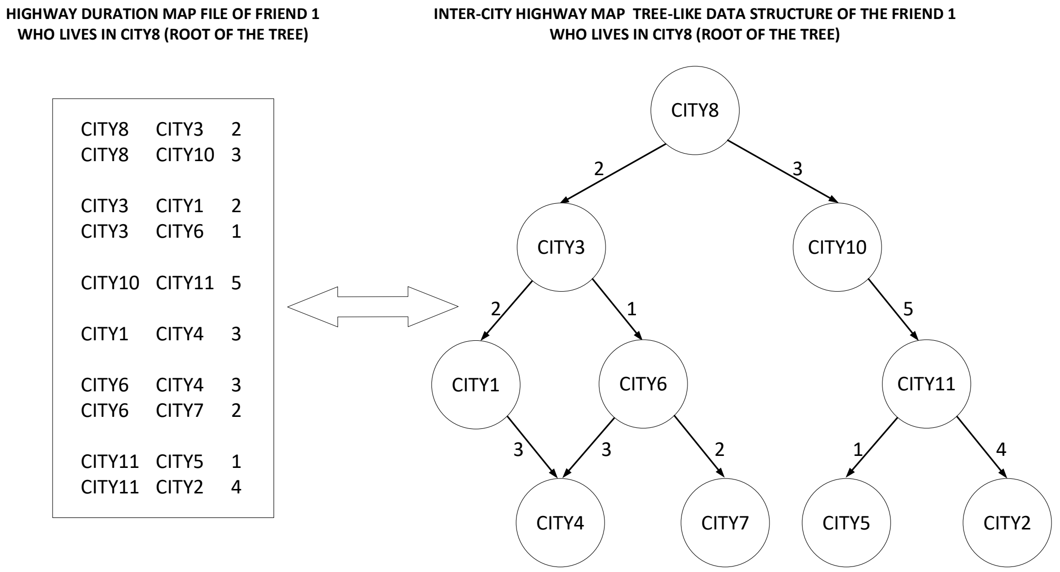
The requested steps for the homework is shown in the following sections with an example use case.

STEP-1: USING FILES FOR TREE-LIKE DATA STRUCTURE CONSTRUCTION

The first step is to construct two **tree-like** data structures from the highway duration map files. You can assume that the highway duration map records are links of a **binary** tree-like data structure: the parent node is the source city and the child node is the destination city for a one-way highway link.

Figure 1 and Figure 2 show examples of highway duration map files for FRIEND-1 and FRIEND-2 alongside of the highway map **tree-like** data structures constructed using the lines in these files. We call the data structure as **tree-like** since there may be more than one paths from a node to another node in the data structure. You will prune all the excess links in the STEP-2 of this homework to make it a tree.

Please note that, the duration maps of two friends may contain different durations for the same city pair (one friend may drive faster while the other one drives slower). Therefore, construct the tree-like data structures based on the given map files separately.

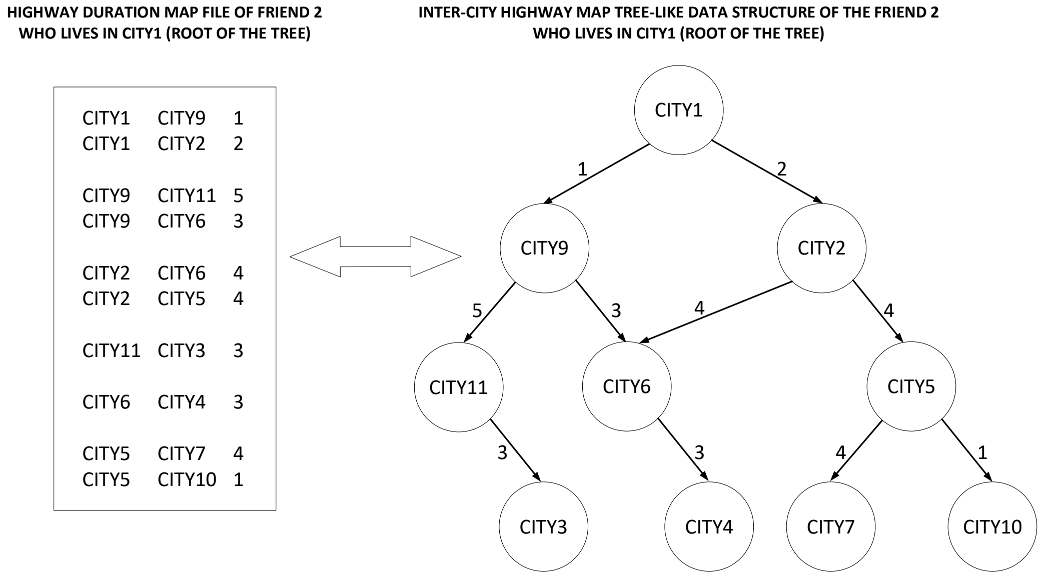


HIGHWAY DURATION MAP FILE OF FRIEND 1
WHO LIVES IN CITY8 (ROOT OF THE TREE)

```
CITY8    CITY3    2
CITY8    CITY10   3

CITY3    CITY1    2
CITY3    CITY6    1

CITY10   CITY11   5

CITY1    CITY4    3

CITY6    CITY4    3
CITY6    CITY7    2

CITY11   CITY5    1
CITY11   CITY2    4
```

INTER-CITY HIGHWAY MAP TREE-LIKE DATA STRUCTURE OF THE FRIEND 1
WHO LIVES IN CITY8 (ROOT OF THE TREE)

Figure 1: City Highway Map File of FRIEND-1 (LEFT) and the City Highway Map Tree-like data structure constructed from the file (RIGHT).



HIGHWAY DURATION MAP FILE OF FRIEND 2
WHO LIVES IN CITY1 (ROOT OF THE TREE)

```
CITY1    CITY9    1
CITY1    CITY2    2

CITY9    CITY11   5
CITY9    CITY6    3

CITY2    CITY6    4
CITY2    CITY5    4

CITY11   CITY3    3

CITY6    CITY4    3

CITY5    CITY7    4
CITY5    CITY10   1
```

INTER-CITY HIGHWAY MAP TREE-LIKE DATA STRUCTURE OF THE FRIEND 2
WHO LIVES IN CITY1 (ROOT OF THE TREE)

Figure 2: City Highway Map File of FRIEND-2 (LEFT) and the City Highway Map Tree-like data structure constructed from the file (RIGHT).

STEP-2: PRUNING THE TREE-LIKE DATA STRUCTURES

In the constructed tree-like data structure, there may be more than one paths with different duration costs from the root of the tree to a city. Consider Figure 1 in which there are two paths between CITY8 to CITY 4:

1) CITY8->CITY3->CITY1->CITY4 which has a duration cost of 7 (2+2+3)
2) CITY8->CITY3->CITY6->CITY4 which has a duration cost of 6 (2+1+3)

Therefore, the link between CITY1 and CITY4 is an excess link and must be pruned. Similarly, the link between CITY2 and CITY6 is an excess link in Figure 2 and must be pruned.

Figure 3 shows the inter-city highway map **trees** of FRIEND-1 and FRIEND-2 which are pruned tree-like data structures of FRIEND-1 and FRIEND-2.

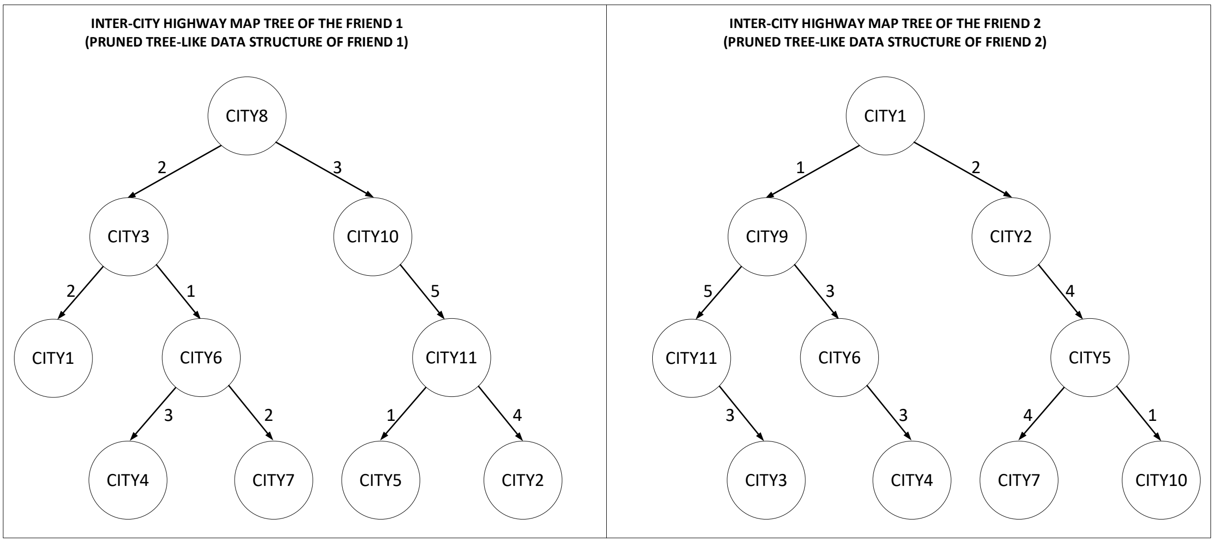

INTER-CITY HIGHWAY MAP TREE OF THE FRIEND 1
(PRUNED TREE-LIKE DATA STRUCTURE OF FRIEND 1)
CITY8
2
3
CITY3
CITY10
2
1
5
CITY1
CITY6
CITY11
3
2
1
4
CITY4
CITY7
CITY5
CITY2
INTER-CITY HIGHWAY MAP TREE OF THE FRIEND 2
(PRUNED TREE-LIKE DATA STRUCTURE OF FRIEND 2)
CITY1
1
2
CITY9
CITY2
5
3
4
CITY11
CITY6
CITY5
3
3
4
1
CITY3
CITY4
CITY7
CITY10

Figure 3: Highway Map Tree for FRIEND-1 (LEFT) and FRIEND-2 (RIGHT)

STEP-3: FINDING THE MEETING POINT AND MINIMUM DURATION

You have to search for the meeting point that minimizes the total journey duration by exploring each possible meeting point. The total duration cost is the sum of the duration cost of for FRIEND-1 to the meeting point and the duration cost of FRIEND-2 to the meeting point.

For example, consider all possible meeting points and their total costs, taking each node of the tree for FRIEND-1 as meeting point level by level:

- CITY8: The duration cost of FRIEND-1 is 0 (FRIEND-1 waits for FRIEND-2 where (s)he lives) and duration cost of FRIEND-2 is INFINITE (there is no path from CITY1 to CITY8 in the tree of FRIEND-2). Therefore, it is not possible to meet at CITY8.

- CITY3: The duration cost of FRIEND-1 is 2 (CITY8->CITY3) and duration cost of FRIEND-2 is 9 (CITY1->CITY9->CITY11->CITY3). Therefore, total duration cost is 11 (2+9) when CITY3 is chosen as the meeting point.

- CITY10: The duration cost of FRIEND-1 is 3 (CITY8->CITY10) and duration cost of FRIEND-2 is 7 (CITY1->CITY2->CITY5->CITY10). Therefore, total duration cost is 10 (3+7) when CITY10 is chosen as the meeting point.
- CITY1: The duration cost of FRIEND-1 is 4 (CITY8->CITY3->CITY1) and duration cost of FRIEND-2 is 0 (FRIEND-2 waits for FRIEND-1 where (s)he lives). Therefore, total duration cost is 4 (4+0) when CITY1 is chosen as the meeting point.
- CITY6: The duration cost of FRIEND-1 is 3 (CITY8->CITY3->CITY6) and duration cost of FRIEND-2 is 4 (CITY1->CITY9->CITY6). Therefore, total duration cost is 7 (3+4) when CITY6 is chosen as the meeting point.
- CITY11: The duration cost of FRIEND-1 is 8 (CITY8->CITY10->CITY11) and duration cost of FRIEND-2 is 6 (CITY1->CITY9->CITY11). Therefore, total duration cost is 14 (8+6) when CITY11 is chosen as the meeting point.
- CITY4: The duration cost of FRIEND-1 is 6 (CITY8->CITY3->CITY6->CITY4) and duration cost of FRIEND-2 is 7 (CITY1->CITY9->CITY6->CITY4). Therefore, total duration cost is 13 (4+7) when CITY4 is chosen as the meeting point.
- CITY7: The duration cost of FRIEND-1 is 5 (CITY8->CITY3->CITY6->CITY7) and duration cost of FRIEND-2 is 10 (CITY1->CITY2->CITY5->CITY7). Therefore, total duration cost is 15 (5+10) when CITY7 is chosen as the meeting point.
- CITY5: The duration cost of FRIEND-1 is 9 (CITY8->CITY10->CITY11->CITY5) and duration cost of FRIEND-2 is 6 (CITY1->CITY2->CITY5). Therefore, total duration cost is 15 (9+6) when CITY5 is chosen as the meeting point.
- CITY2: The duration cost of FRIEND-1 is 12 (CITY8->CITY10->CITY11->CITY2) and duration cost of FRIEND-2 is 2 (CITY1->CITY2). Therefore, total duration cost is 14 (12+2) when CITY2 is chosen as the meeting point.

Considering all the possibilities above, CITY1 is the meeting point that minimizes total duration cost.

EXPECTED OUTPUT

You have to **construct the tree-like data structures and prune them to get the highway map trees** as we described in STEP-1 and STEP-2. Then you are expected to **output highway map trees (e.g., pruned tree-like data structures) for FRIEND-1 and FRIEND-2** by using the **pre-order** tree traversal as follows:

```
FRIEND-1: CITY8 CITY3 CITY1 CITY6 CITY4 CITY7 CITY10 CITY11 CITY5 CITY2
FRIEND-2: CITY1 CITY9 CITY11 CITY3 CITY6 CITY4 CITY2 CITY5 CITY7 CITY10
```

Then, you are expected to output meeting point and total duration cost as follows:

```
MEETING POINT: CITY1
TOTAL DURATION COST: 4
```

SUBMISSION RULES

We opened a message board “**Project 2 Q&A**” on Ninova. First of all, check this message board if you have any questions; it could be asked before and answered. If you can't find the answer for your question, then write it to the message board so that other students can see and benefit from it.

Make sure you wrote your name and student number within the first lines of all your project files, in the following format:

```
/* @Author
Student Name: <student_name>
Student ID: <student_id>
E-mail: <e-mail address>
Date: <date>
*/
```

Use comments wherever necessary in your code to explain what you did.

You should upload your work to Ninova before the given deadline, otherwise your work will not be graded. Do not leave the submission to the last minute! Give yourself at least 15 minutes before the deadline to upload your work so that you have enough time to upload it before the deadline expires. **Please don’t forget, uploading the homework also takes some time.**

You may discuss the problems at an abstract level with your classmates, but **you should not share or copy code from your classmates or from the Internet**. You should submit your own, individual homework. Academic dishonesty, including cheating, plagiarism, and direct copying, is unacceptable. Note that **YOUR CODE WILL BE CHECKED WITH THE PLAGIARISM TOOLS!**