

Report of Homework 2

BLG 354E-Analysis of Algorithms

Name and surname: Jilan Alrehaili

Student number: 150160922



22.Apr.2019

Department of Computer Engineering

constructGraph method:

construct value from input values. Values will construct the graph according to input value. On (if counter==1) first line of the text file read the number of nodes and construct the row and column of each line on the text file. (if counter==2) takes source and target. Else it will read the rest of the edges.

minDistance method:

its a helper function that we will use on dijkstra method to achieve the shortest distance between the visited list.

dijkstra method:

iterate through the graph according to the edges and set to the number it corresponds to or infinity. And check the minimum distance. At the end print the total duration time while looping from distant target.

checkPath method:

check if the road that was deleted makes errors or not according to the outcome that this method will receive from dijkstra method.

DoDijkstra method:

our 4 stages consist of going twice and returning twice for every person. So we set the second target to false to deal with the first stage and call the dijkstra method inside it. Iterate through the graph and sum the paths and store it on p1. And do the same for second stage. And after that I will test the go through the paths action. If the first path and second has the same value and node then we will set a flag true to stop and break from the loop and see which path I will change. If both flags have no problem but the total of second stage is more than the total of first stage then change the path of second stage. but if the flag is 0 for any of them then change path for the one with the 0 flag. If both are 0 then its an error. Now into third stage with it return stage. we set return as false and set the target as source and source as target. And do the same as before but increase the first node with +30. and the fourth is the same and test like the one before.

Main function:

it calls constructGraph and doDijkstra methods to start the dijkstra algorithm and find the shortest path and print the results.

RUN THE CODE USING THOSE TWO LINE:

- g++ -o hw hw2.cpp

- ./hw <txt file>

Testing text file #1:

```
>> Path from node 0 to 5 <<
Node 0 to 1 Time: 4
Node 1 to 4 Time: 7
Node 4 to 5 Time: 20
(Node 5 wait: 50)
---- return ----
Node 5 to 6 Time: 56
Node 6 to 2 Time: 58
Node 2 to 3 Time: 68
Node 3 to 1 Time: 73
Node 1 to 0 Time: 79
Total duration time is: 79

>> Path from node 2 to 4 <<
Node 2 to 3 Time: 10
Node 3 to 1 Time: 15
Node 1 to 4 Time: 18
(Node 4 wait: 48)
---- return ----
Node 4 to 3 Time: 49
Node 3 to 1 Time: 54
Node 1 to 0 Time: 60
Node 0 to 2 Time: 68
Total duration time is: 68
```

Testing text file #2:

```
>> Path from node 0 to 9 <<
Node 0 to 2 Time: 5
Node 2 to 1 Time: 7
Node 1 to 6 Time: 11
Node 6 to 7 Time: 13
Node 7 to 9 Time: 21
(Node 9 wait: 51)
---- return ----
Node 9 to 10 Time: 54
Node 10 to 6 Time: 59
Node 6 to 3 Time: 60
Node 3 to 1 Time: 67
Node 1 to 0 Time: 70
Total duration time is: 70

>> Path from node 3 to 15 <<
Node 3 to 10 Time: 8
Node 10 to 6 Time: 13
Node 6 to 7 Time: 15
Node 7 to 8 Time: 18
Node 8 to 11 Time: 20
Node 11 to 15 Time: 25
(Node 15 wait: 55)
---- return ----
Node 15 to 16 Time: 64
Node 16 to 14 Time: 72
Node 14 to 5 Time: 83
Node 5 to 10 Time: 87
Node 10 to 6 Time: 92
Node 6 to 3 Time: 93
Total duration time is: 93
```

Testing text file #3:

```
>> Path from node 0 to 6 <<
Node 0 to 3 Time: 4
Node 3 to 2 Time: 13
Node 2 to 4 Time: 18
Node 4 to 6 Time: 31
(Node 6 wait: 61)
---- return ----
Node 6 to 3 Time: 65
Node 3 to 5 Time: 71
Node 5 to 1 Time: 78
Node 1 to 0 Time: 84
Total duration time is: 84

>> Path from node 2 to 1 <<
Node 2 to 4 Time: 5
Node 4 to 5 Time: 10
Node 5 to 1 Time: 17
(Node 1 wait: 47)
---- return ----
Node 1 to 0 Time: 53
Node 0 to 3 Time: 57
Node 3 to 2 Time: 66
Total duration time is: 66
```

Testing text file #4:

```
>> Path from node 4 to 5 <<
Node 4 to 1 Time: 7
Node 1 to 2 Time: 11
Node 2 to 5 Time: 14
(Node 5 wait: 44)
---- return ----
Node 5 to 3 Time: 53
Node 3 to 6 Time: 58
Node 6 to 4 Time: 64
Total duration time is: 64

>> Path from node 0 to 7 <<
Node 0 to 3 Time: 5
Node 3 to 6 Time: 10
Node 6 to 4 Time: 16
Node 4 to 7 Time: 26
(Node 7 wait: 56)
---- return ----
Node 7 to 6 Time: 59
Node 6 to 0 Time: 67
Total duration time is: 67
```

Testing text file #5:

```
Error! There is no solution
```