

2011.2 Object Oriented Programming Design, Project #2

(Due : Oct. 17th 11:59pm)

Submission Rule

1. Create a directory "proj2" and its subdirectories "Prob1" and "Prob2" in "proj2"
2. Insert compilable source code package for Problem#1 into "Prob1" and source code package for Problem#2 into "Prob2"
3. zip the directory "proj2" into proj2.zip and submit the zip file into eClass homework board(과제방).

Problem1. Assume a wireless network system where wireless communication can be done only between mobile phones. Each mobile phone is located in a two dimensional space, and has x (horizontal) and y (vertical) coordinate values. Due to the limitation of the wireless communication, each mobile phone have direct connection only to the mobile phones located within the euclidean distance d. If mobile phones A and B have direct connection, and B and C have direct connection, then we say A and C have indirect connection. If two mobile phones have direct or indirect connection, the two mobile phones can communicate with each other.

This kind of a wireless network system can be represented by a graph. Each mobile phone is a node. If any two mobile phones have direct connection, then the two nodes(mobile phones) are linked by an edge.

Write a C++ program that takes a set of mobile phones coordinates in a wireless network system and two selected mobile phones from an input file, transforms the network system into a graph, and computes the shortest path between the two nodes that are corresponding to the two selected mobile phones. The program should print the shortest path to the screen as an output. If there are more than one shortest paths that have the same length, print all the shortest paths as output. You may ignore the efficiency of your program.

Note.

Here, the length of a path between two nodes A and B is defined as the number of edges when traversing from A to B. The shortest path means the path with minimum length. There cannot be two or more same nodes in a path. (that means there should be no cycle in a path.)

Input Format : file input (input.txt)

The first line has the number of mobile phones N ($N \leq 100$) and the distance d. In the second line, the index numbers of two selected mobile phones are given. Assume that index number starts from 1, not 0. From the third line, each line has the x and y coordinate values of mobile phones in order starting from #1 mobile phone to #N mobile phone.

Output Format : screen output

Print out the shortest path(s) (a list of node index numbers) between the two selected mobile phone nodes that are given as input.

Execution example

```
> prob1.exe input.txt
```

Input example (input.txt)

```
10 2.5
1 6
0.3 1.2
1.5 2.3
4.5 5.2
1.2 0.9
3.1 -1.2
3.5 -2.3
5.1 3.9
1.9 2.1
2.0 0.1
5.1 6.1
```

Output example (screen output)

```
1956
```

Problem2. Complete implementing `inf_int` class that represents infinite precision integer and its operations.

the specification of `inf_int` is provided to you in "`infinite_int.h`". You should add its implementation in "`infinite_int.cpp`". Test the correctness of your implementation by writing `main.cpp`. "`infinite_int.h`" and an example of "`main.cpp`" are downloadable from our class website. After compilation, your code should generate correct result. The result of this problem should be submitting compilable source code package (`infinite_int.h` , `infinite_int.cpp` , `main.cpp` , `.sln` files (files for opening with visual studio 2008)). FYI, I will test your code by replacing the "`main.cpp`" with my own `main.cpp` files.