

# All-pairs shortest path

## Problem description

Given a weighted directed graph, you need to compute the shortest path between any two nodes. You also need to print the shortest path.

## Input/ output

First line of the input has three integer numbers  $N$ ,  $E$ ,  $Q$  where  $N$  is the number of nodes,  $E$  is the number of edges, and  $Q$  is the number of queries. Nodes are numbered as 1, 2, ...,  $N$ .

Each of the following  $E$  lines describes an edge. Each edge is represented by three integers  $A B C$  (i.e., the cost of going **from**  $A$  **to**  $B$  is  $C$ ) where  $1 \leq A, B \leq N$  and  $N \leq 100$ .

Each of the next  $Q$  lines contains a query. Each query is represented by two integers  $A B$ . *You need to find the shortest path from  $A$  to  $B$ . If there is a path from  $A$  to  $B$ , print the cost of the shortest path and **the actual path**. If there is no path from  $A$  to  $B$ , just print **NO PATH**.*

**See sample I/O for clarification.**

## Sample input/ output

Sample input	Sample output
<b>5</b> 7 3 1 2 5 1 3 2 2 3 -2 3 4 7 2 4 3 5 2 9 4 5 10 1 5 <b>5</b> 1 1 1	<b>cost = 18</b> <b>1-2-4-5</b> <b>NO PATH</b> <b>cost = 0</b> <b>1-1</b>

## Deliverables:

1. Makefile
2. Your source code

### Notes:

1. Make sure that your code runs on the CISE machines.
2. Name the executable as “**Floyd-Warshall**” in your submitted Makefile.
3. Do not compress your submission. Just submit the Makefile and source file.
4. Follow the output format **exactly**. Do not print anything extra (e.g., prompt, whitespaces etc.).
5. Use `cin>>`, `cout <<` (`scanf`, `printf`) for I/O. Assume the input from standard input and write to standard output.
6. If there are multiple paths with the same shortest path cost, print any of them.

### Points:

Criteria	Points
Correct output	5
Code organization	2
Comments	3
Total	10

### Sample Input Graph:

The shortest path from **1** to **5** is **1-2-4-5** and the cost is 18.

