

# "Shortest" pair of paths

A chemical company has an unusual shortest path problem. There are  $N$  depots (vertices) where chemicals can be stored. There are  $M$  individual shipping methods (edges) connecting pairs of depots. Each individual shipping method has a cost. In the usual problem, the company would need to find a way to route a single shipment from the first depot (0) to the last ( $N - 1$ ). That's easy. The problem they have seems harder. They have to ship two chemicals from the first depot (0) to the last ( $N - 1$ ). The chemicals are dangerous and cannot safely be placed together. The regulations say the company cannot use the same shipping method for both chemicals. Further, the company cannot place the two chemicals in same depot (for any length of time) without special storage handling --- available only at the first and last depots.

To begin, they need to know if it's possible to ship both chemicals under these constraints. Next, they need to find the least cost of shipping both chemicals from first depot to the last depot. In brief, they need two completely separate paths (from the first depot to the last) where the overall cost of both is minimal.

Your program must simply determine the minimum cost or, if it's not possible, conclusively state that the shipment cannot be made.

The input will consist of multiple cases. The first line of each input will contain  $N$  and  $M$  where  $N$  is the number of depots and  $M$  is the number of individual shipping methods. You may assume that  $N$  is less than 20 and that  $M$  is less than 200.

The next  $M$  lines will contain three values,  $i$ ,  $j$ , and  $v$ . Each line corresponds a single, unique shipping method. The values  $i$  and  $j$  are the indices of two depots, and  $v$  is the cost of getting from  $i$  to  $j$ .

Note that these shipping methods are directed. If something can be shipped from  $i$  to  $j$  with cost 10, that says nothing about shipping from  $j$  to  $i$ . Also, there may be more than one way to ship between any pair of depots, and that may be important here.

A line containing two zeroes signals the end of data and should not be processed.

## Sample

### Input:

```
2 1
0 1 20
2 3
0 1 20
0 1 20
1 0 10
4 6
0 1 22
1 3 11
0 2 14
2 3 26
0 3 43
0 3 58
0 0
```

**Output:**

Instance #1: Not possible

Instance #2: 40

Instance #3: 73