You are a political prisoner in jail. Things are looking grim, but fortunately, your jailmate has come up with an escape plan. He has found a way for both of you to get out of the cell and run through the city to the train station, where you will leave the country. Your friend will escape first and run along the streets of the city to the train station. He will then call you from there on your cellphone (which somebody smuggled in to you inside a cake), and you will start to run to the same train station. When you meet your friend there, you will both board a train and be on your way to freedom.

Your friend will be running along the streets during the day, wearing his jail clothes, so people will notice. This is why you can not follow any of the same streets that your friend follows - the authorities may be waiting for you there. You have to pick a completely different path (although you may run across the same intersections as your friend).

What is the earliest time at which you and your friend can board a train?

Problem, in short

Given a weighed, undirected graph, find the shortest path from **S** to **T** and back without using the same edge twice.

Input

The input will contain several test cases. Each test case will begin with an integer n ($2 \leq n \leq 100$) — the number of nodes (intersections). The jail is at node number 1, and the train station is at node number n. The next line will contain an integer m — the number of streets. The next m lines will describe the m streets. Each line will contain 3 integers — the two nodes connected by the street and the time it takes to run the length of the street (in seconds). No street will be longer than 1000 or shorter than 1. Each street will connect two different nodes. No pair of nodes will be directly connected by more than one street. The last test case will be followed by a line containing zero.

Output

For each test case, output a single integer on a line by itself — the number of seconds you and your friend need between the time he leaves the jail cell and the time both of you board the train. (Assume that you do not need to wait for the train — they leave every second.) If there is no solution, print 'Back to jail'.

Sample Input

```
2
1
1 2 999
3
3
1 3 10
2 1 20
3 2 50
9
12
1 2 10
1 3 10
1 4 10
2 5 10
3 5 10
4 5 10
5 7 10
6 7 10
7 8 10
6 9 10
7 9 10
8 9 10
0
```

Sample Output

```
Back to jail
80
Back to jail
```