Problem COOLLATE: Cool People Are Always Late

As you might know, the cool people are never on time when they join a party. Instead, they prefer to be among the last that arrive at the party. That makes them important.

Today, you want to be important. Unfortunately, you are quite bored just sitting at home and doing nothing because your computer is in repair. So, you eventually decide to leave your apartment. The only problem is that you would definitely not like to be the first to arrive at the place, since that would show everybody that you have got no friends and are a really miserable being. On the other hand, you do not want to be too late, either, because you are not cool enough yet.

Your task is, given a map from your city, to find the second-shortest route from your home to the party. This is because the shortest route would let you arrive too early, while taking any route longer than the second-shortest is still too cool for you. You will, of course, not have any cycles in your route, because that would be really uncool (a cycle in this sense is any route on which the same place is visited more than once).

It is guaranteed that there is exactly one route of shortest length.

Input

The first line of the input file contains a single integer n, the number of test cases that will follow. The next line is blank.

For each test case, there is one line with two integers v and e, the number of places (vertices) on the map and the number of streets (edges) connecting them, respectively. The next e lines contain the description of a street, consisting of three integers a, b and len. The line "3 8 10" would mean that there is a street of length 10 connecting the places 3 and 8. Each test case is finished by a line with two integers h and p, the place of your home and the one of the party, followed by a blank line.

There are no one-way streets, and for every pair of places (a, b) there is at most one street connecting them. No street has a negative length, and no street is longer than 1000. The number of places will not exceed 100, the number of streets will not exceed $1,000$.

Output

Print one line for every test case. The line should contain the length of the shortest path and, if it exists, the length of the second shortest path.

Sample Input 1

```
3

4 4
0 1 10
0 3 5
1 2 10
3 1 8
0 2

4 3
0 1 10
1 2 10
2 3 10
0 3

5 6
0 1 8
0 3 5
1 2 7
1 4 5
3 1 5
4 2 4
0 2
```

Sample Output 1

```
20 23
30
15 17
```