

Almost everything is unique in Unique world. The unique creature, unika lives in this world. Each unika has its own unique house. All the houses are connected by roads in such a way so that there is exactly one (unique !!) path from one house to another one. More importantly the cost to traverse each road is unique. In a recent observation, Xenique (President of Unique world) found that the costs of all possible paths are not unique and this should not happen in the Unique World. So, he decided that the cost of each path would be unique. To make this, he also decided that unika can traverse same road more than once while going from one house to another one except the last road on the path. But soon or later he was informed that too many traffic jam is being created due to this rule. Xenique called a meeting on this issue.

A young unika, Prognique suggested that they should use minimum number of roads while visiting from one place to another one. He also told that they should use only those roads, which are necessary for the path. And only a computer program can help in this case.

Input

The first line of the input file contains a single integer N ($0 < N \leq 20$) that denotes the number of inputs. Each data set starts with two positive integers n ($n < 51$), which denotes the number of unika and m , the number of roads in the unique world. Each of next m lines contains three positive integers, the id of unika (Each unika has a unique id between 1 to n) id_1 , id_2 and c . There is a road of traversing cost c connecting the house of these two unika. Next line contains another integer k ($1 \leq k \leq 20$), the number of queries. Each of next k lines contains three integers the id of the unika of the source and destination houses and the required $cost$ (between 1 and 100000) of this path. Input may contain blank lines.

Output

For each data set, if it is possible to traverse the path using the given cost, print 'Yes' and the minimum number of roads. Otherwise just print 'No'. Put a blank line between two consecutive sets of inputs.

Sample Input

```
1
5 4
1 2 2
1 3 3
1 4 4
1 5 5
5
1 2 2
2 3 5
2 3 6
4 5 10
2 4 18
```

Sample Output

```
Yes 1
Yes 2
No
No
Yes 8
```