# Winter Crush

Winter is a sad season, specially if you have nobody to share your time with. Valentine's day is coming and Dickie is crushed for Elizabeth, but she has already friendzoned him with the classic "You're my best friend" and other silly phrases used by girls in order to keep the "friend" neutralized but at the same time, keep him close.

Nevertheless, Dickie ain't gonna to take any more stools. He's thinking about an ultimate move to gain Elizabeth heart! He has called this "No more stools operation!", and will take place on Valentine's day. He hasn't clearly decided what to do, because that heavily depends on Elizabeth's position on Valentine's day. Even so, Dickie has been smart enough to figured out a way to know Elizabeth's position in terms of probability based on the time elapsed from the beginning of the day.

As Elizabeth's "best friend", Dickie knows her preferences and tastes. He knows that if she's at a given position x, she will move to any adjacent position y with a probability $p_{xy}$ depending on the joy attained going there and the amount of snow on the path. The joy $j_{xy}$ is being measured in joysons and the snow $s_{xy}$ by centimeters, and the choice will be taken by the relation $j_{xy} / s_{xy}$. Some properties shall hold:

- The set of all positions is to be named V.
- For any given x ∈ V, the set of all positions y ∈ V adjacent to x is to be named Ex. A position y is to be called adjacent if there is a path from x to y. Paths are bidirectional, so $j_{xy}/s_{xy} = j_{yx}/s_{yx}$.
- ∀x ∈ V ∧ ∀y ∈ Ex:

$$p_{xy} = \frac{j_{xy}/s_{xy}}{\sum_{y \in E_x} j_{xy}/s_{xy}}$$

$$\sum_{p_{xy}} p_i = 1$$

$$p_{xx} = 0$$

$$p_{xy} = p_{yx}$$

$$\forall z \in V \land \notin E_x \Rightarrow p_{xz} = 0$$

Dickie has some places in mind where he can find Elizabeth, but he's aware that she could depart that morning from either her house or any of hers friends' houses, so he wants to know which is the position with the highest probability to find Elizabeth given an elapsed time and the position she's departing. Dickie wants to evaluate several possible scenarios, so a query will be formed by a time Ti and several possible initial positions Pi.

## Input

The first line contains an integer C, which specifies the number of test cases. Then, will follow the descriptions of C test cases.

Each case will start with two integers, N and M, denoting the number of positions and the number of paths respectively. Next M will describe each of the paths between positions with four integers x, y, $j_{xy}$, $s_{xy}$, with the significance given above in the statement. All position indexes are 0-based.

The line M + 1 will contain an integer Q denoting the number of Dickie's queries. Each query will start with two integers, Ti and qi denoting the elapsed time and the number Elizabeth's possible starting positions. Next qi lines will contain an integer will represent Elizabeth's starting positions Pi.

## Output

For each test case you should print the string "Scenario #i:" where i represents the test case you are analyzing (starting from 1), followed by a blank line. For each query Qi in the input, you must output a line with qi integers representing the position with the highest probability to find Elizabeth given the elapsed time Ti and the starting position Pi of her. If there are several positions with the highest probability, print the smallest one. If there are no positions with higher probability than 0, print -1.

## Sample

**Input:**
```
2
3 3
0 1 10 2
0 2 1 8
1 2 100 1
2
1 1 0
2 1 0
4 5
0 1 10 2
1 3 2 3
0 2 4 8
2 1 1 10
2 3 1 1
2
2 1 2
10 2 0 3
```

**Output:**
```
Scenario #1:

1
2

Scenario #2:

1
0 0
```

## Constraints

- $0 < j_{xy}, s_{xy} \le 100$
- $0 \le M \le (N * (N - 1)) / 2$

**Subtask 1 (30%)**

- $1 \le C \le 50$

- $1 \le N \le 20$
- $0 \le |E_x| \le 3, \forall x \in V$
- $1 \le T_i \le 20$
- $1 \le Q \le 10$
- $q_i = 1$
- $0 \le P_i \le N$

**Subtask 2 (70%)**

- $1 \le C \le 20$
- $1 \le N \le 100$
- $1 \le Q \le 10$
- $2 \le T_i \le 2000$
- $1 \le q_i \le 100$
- $0 \le P_i \le N$