8117 Security Badge

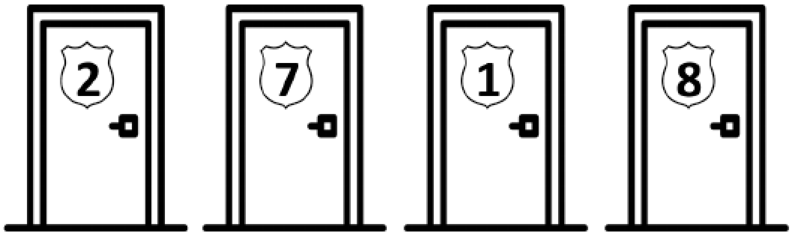
The home office of Labyrinthian Inc. has installed a new system of security badges and electronic door locks. Each badge is assigned an ID number, and the idea was that electronic locks on each door should allow access only to personnel whose ID number indicated that they had appropriate security clearance to enter that room, hallway, closet, or whatever lay on the other side of the door.

The contract for the lock system, however, was put out to the lowest bidder, who clearly misunderstood the intention. Instead of each lock storing a list of permitted ID numbers, instead each lock stores exactly two numbers, a lower and upper bound, and permits passage to badges whose number lies between those bounds. For example, a lock keyed as (25, 29) would pass only badges 25, 26, 27, 28, and 29.

Complicating the matter is the fact that lock on each side of the door can be keyed differently, so a person who is able to pass through the door in one direction might not be able to return once the door has closed behind them.

The results have been frustrating (and occasionally entertaining-videos of everyone in the company trying to find a way to the cafeteria at noon have gone viral on social media).

It has become a major task, when hiring or promoting any employee, to find a badge number that will actually get them from the front door to their new office.



You are in charge of the security for the home office, with n rooms and m doors between the rooms. The rooms and doors are conveniently numbered from 1 to n, and from 1 to m, respectively.

Door i opens between rooms a_i and b_i, maybe with different keys on each side. Additionally, each door has a security code that can be represented as a range of numbers $[c_i, d_i]$.

There are k employees working in the building, each carrying a security badge with a unique, integer-valued badge ID between 1 and k. An employee is cleared to go through door i only when the badge ID x satisfies $c_i \le x \le d_i$.

Your boss wants a quick check of the security of the building. Given s and t, how many employees can go from room s to room t?

Input

The input file contains several test cases, each of them as described below.

The first line of input contains three space-separated integers n, m, and k ($2 \le n \le 1,000$; $1 \le m \le 5,000$; $1 \le k \le 10^9$).

The second line of input contains two space-separated integers s and t ($1 \le s, t \le n$; $s \ne t$).

Each of the next m lines contains four space-separated integers a, b, c, and d indicating that a lock permits passage from room a to room b (but not from b to a) for badges numbered from c to d, inclusive. It is guaranteed that $1 \le a, b \le n$, $a \ne b$, $1 \le c \le d \le k$, and no (a, b) pair will occur more than once, although both (a, b) and (b, a) may occur within separate lines.

For any given pair of rooms a, b there will be at most one door from a to b (but there may be both a door from a to b and a door from b to a).

Output

For each test case, print, on a single line, the number of employees who can reach room t starting from room s.

Sample Input

```
4 5 10
3 2
1 2 4 7
3 1 1 6
3 4 7 10
2 4 3 5
4 2 8 9
4 5 9
1 4
1 2 3 5
1 3 6 7
1 4 2 3
2 4 4 6
3 4 7 9
```

Sample Output

```
5
5
```