

Little Mika is at the North Pole and is standing on one of the many icebergs. He wants to go meet his friend Laza, who is standing on another iceberg. The water between certain icebergs is passable and can either be completely liquid, or full of tiny ice chunks that make the trip difficult. Mika has a machine that can melt the tiny chunks (but not entire icebergs), which is **initially off**. He can switch it on and off as many times as he wants.

While the machine is on, he can cross gaps of water between icebergs that are filled with ice chunks, which mysteriously reappear after he crosses. He must not cross gaps that have no ice in this case, because that would risk damage to the machine, so he would have to turn the machine off (and keep it off until he encounters ice chunks again).

You should find a way for Mika to meet Laza that **minimises the number of times the machine is switched on or off**. The machine can be on when he meets Laza, and he doesn't have to switch it off at the end.

Input format

The first line of the standard input contains the number of icebergs N and the number of paths between them M (it might be impossible to cross between certain pairs, because the way is too difficult).

The next M lines describe one path each. Each line contains space-separated integers a , b and t , meaning that there is a path between icebergs a and b , where t is **1** if the path is filled with ice chunks, and **0** otherwise.

The final line of the input contains two integers u and v : u is the number of the iceberg Mika starts on, and v is the "goal" iceberg where Laza is.

Output format

Your program should print one number to the standard output -- the lowest possible number of times Mika has to turn the machine on or off to reach Laza.

Sample 1

Input

```
...
4 4
1 2 1
1 3 1
2 3 0
3 4 0
1 4
...
```

Output

```
...
2
...
```

Explanation

Mika can go to the iceberg 2 or 3, and has to turn the machine on for either. From either of the two, he can go to 4, and would have to turn the machine off, so the total number of changes is 2.

Sample 2

Input

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

Output

```
...
4
...
```

Explanation

The only path that reaches iceberg 7 is $1 - 3 - 5 - 6 - 7$.

Constraints

- $1 \leq N, M \leq 200000$
- $1 \leq a, b, u, v \leq N$
- $t = 0 \vee t = 1$

Testcases are split into four non-overlapping groups:

- In tests worth 10 points: $M = N - 1$, and there are at most 2 paths leaving each iceberg.
- In tests worth 10 points: all paths are full of ice chunks.
- In tests worth 25 points: $M = N - 1$ and each iceberg is reachable from any other (not necessarily via a direct path).
- In tests worth 30 points: $N, M \leq 5000$.
- In tests worth 25 points: no additional constraints.