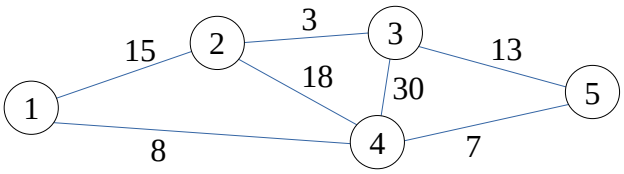
Tunnels

1 second, 256 MB

In an underground city, there is a network of ***M*** bidirectional tunnels joining ***N*** intersections, namely intersections 1, 2,…, and ***N***. Each tunnel ***i*** (for 1<=***i***<=***M***) only allows cars whose height are *at most* H_i. When driving through these tunnels, you have to pay the fee of 1 Baht for each tunnel.

Write a program to find the lowest fee one has to pay to go from intersection ***S*** to intersection ***T*** with a car of height ***C***.

Consider the following example with ***N*** = 5 and ***M*** = 7. The height of each tunnel is shown next to it.



If you want you go from intersection 1 to intersection 5 with a car of height 7, you can go from intersection 1 to intersection 4 and finally to intersection 5, passing through 2 tunnels and pay 2 Baht.

If the height of the car is 10, going from intersection 1 to 5 requires you to go through 4 tunnels (i.e., from intersection 1 to 2, to 4, to 3, and finally to 5); thus you have to pay 4 Baht.

If the height of the car is 15, there is no way to reach intersection 5 from intersection 1.

Input

The first line of input contains two integers ***N*** and ***M*** (2 <= ***N*** <= 3,000; 1 <= ***M*** <= 30,000).

The next **M** lines contains tunnel information. Specifically, for 1 <= ***i*** <= ***M***, line 1+***i*** contains three integers A_i B_i and H_i, that specifies that tunnel ***i*** joins intersections A_i and B_i, and allows cars of height at most H_i (1<=A_i<=***N***; 1<=B_i<=***N***; 1 <= H_i <= 1,000).

The last line contains three integers ***S*** ***T*** and ***C*** (1<=***S***<=***N***; 1<=***T***<=***N***; 1 <= ***C*** <= 1,000)

Output

The output contains one line, specifying the lowest fee you have to pay for a car of height ***C*** to go from intersection ***S*** to ***T***. If it is impossible to do so, you should output `-1`.

Example 1

Input	Output
5 7 1 2 15 3 2 3 4 2 18 3 4 30 3 5 13 4 1 8 4 5 7 1 5 7	2

(More examples on the next page.)

Example 2

Input	Output
5 7 1 2 15 3 2 3 4 2 18 3 4 30 3 5 13 4 1 8 4 5 7 1 5 10	4

Example 3

Input	Output
5 7 1 2 15 3 2 3 4 2 18 3 4 30 3 5 13 4 1 8 4 5 7 1 5 15	-1