CSE221 Assignment 08 Spring 2025

A. Friendship

1 second, 1024 megabytes

Study material: http://www.shafaetsplanet.com/?p=763

There is a group of $\mathbf{N}$ people living in a small village. Each person in the village has a unique identity — labeled with an integer value between 1 and N. Initially, the villagers don't have any friends. As time passes, they begin to form friendships.

In this problem, you will be given information of $\mathbf{K}$ friendships. You have to print an integer value that denotes the size of their friend circle.

Suppose five people are living in the village, labeled 1, 2, 3, 4, and 5. Initially, the size of each friend circle is one, since no friendships have been formed yet. One day, person 1 and person 2 become friends. So, the size of their friend circle becomes two. The next day, person 3 and person 4 become friends, and the size of their friend circle becomes two as well. After a few days, person 1 and person 4 become friends. Now, the size of their combined friend circle becomes four, consisting of persons 1, 2, 3, and 4.

Input

The first line contains two integers N, K $(1 \leq N, K \leq 3 \times 10^5)$ — the total number of people, total number of friendship created.

The next K lines contain two integers $a_i, a_j (1 \leq a_i, a_j \leq N, a_i \neq a_j)$ — two people become friends.

Output

For each friendship formation, output a single integer on a new line representing the size of the friend circle that the two people belong to after becoming friends.

input
```
8 7
2 4
4 5
3 6
4 7
3 1
2 7
6 2
```
output
```
2
3
2
4
3
4
7
```

input
```
6 5
4 6
2 6
1 6
1 4
1 2
```
output
```
2
3
4
4
4
```

input
```
2 1
1 2
```

output

```
2
```

input

```
5 7
4 5
3 5
3 4
2 5
2 3
1 5
1 4
```

output

```
2
3
3
4
4
5
5
```

In input sample 1,

Initially, 8 people in the village do not know each other.

{1} {2} {3} {4} {5} {6} {7} {8}

— After person 2 and person 4 become friends:

{1} {2,4} {3} {5} {6} {7} {8}

The output is 2, since the size of the friends circle {2,4} is 2.

— After person 4 and person 5 become friends:

{1} {2,4,5} {3} {6} {7} {8}

The output is 3, since the size of the friends circle {2,4,5} is 3.

— After person 3 and person 6 become friends:

{1} {2,4,5} {3,6} {7} {8}

The output is 2, since the size of the friends circle {3,6} is 2.

— After person 4 and person 7 become friends:

{1} {2,4,5,7} {3,6} {8}

The output is 4, since the size of the friends circle {2,4,5,7} is 4.

— After person 3 and person 1 become friends:

{2,4,5,7} {1,3,6} {8}

The output is 3, since the size of the friends circle {1,3,6} is 3.

Since person 2 and person 7 are already in the same friend circle, nothing changes:

{2,4,5,7} {1,3,6} {8}

The output is 4, since the size of the friends circle {2,4,5,7} is 4.

— After person 6 and person 2 become friends:

{2,4,5,7,1,3,6} {8}

The output is 7, since the size of the friends circle {2,4,5,7,1,3,6} is 7.

B. Help the King!

1 second, 1024 megabytes

In the kingdom of Beluga, there are $\mathbf{N}$ cities connected by $\mathbf{M}$ bidirectional roads. Each road has a maintenance cost. There is at least one way to travel between any two cities.

The king is worried about the growing cost of maintaining all these roads. To fix this, he asks his advisors for help.

The council suggests keeping only the roads needed to connect all the cities with the least total maintenance cost. Instead of building new roads, the king decides to save money by removing some of the existing ones.

Since you're known for your programming skills, the king calls on you. He wants you to figure out the minimum total maintenance cost that can be achieved by removing some roads—while still making sure that all the cities remain connected.

Input

The first line contains two integers $N, M (2 \le N \le 2 \times 10^5, 1 \le M \le 3 \times 10^5)$ — the number of vertices, total number of edges.

The next M lines will contain three integers $u_i, v_i, w_i (1 \le u_i, v_i \le N, 1 \le w_i \le 10^6)$ — there is an edge between the node u_i and the node v_i with a maintenance cost w_i.

Output

The output should contain a single integer, with the minimum total maintenance cost achievable.

input
```
4 3
3 4 5
3 1 5
2 1 2
```
output
```
12
```

input
```
6 5
2 6 3
2 3 3
2 1 4
6 5 1
5 4 2
```
output
```
13
```

input
```
2 1
1 2 9
```
output
```
9
```

input
```
5 7
1 2 8
1 4 3
1 5 4
2 4 8
2 3 5
4 3 4
3 5 5
```
output
```
16
```

C. Again MST

1 second, 256 megabytes

You are given a **bidirectional weighted** graph with N nodes and M edges. The nodes are numbered from 1 to N. The graph contains no self-loops or multiple edges.

Your task is to find the total cost of the second-best Minimum Spanning Tree. If no such tree exists, print -1.

Note: The second-best MST must have a total weight strictly greater than that of the best (minimum) MST.

Input

The first line contains four integers $N, M (2 \le N \le 10^3, 1 \le M \le 2 \times 10^3)$ — the number of vertices, total number of edges.

The next M lines will contain three integers $u_i, v_i, w_i (1 \le u_i, v_i \le N, 1 \le w_i \le 10^6)$ — there is an edge between the node u_i and the node v_i with a weight w_i.

Output

Output the total cost of the second-best Minimum Spanning Tree. If no such tree exists, print -1.

input

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

output

```
18
```

input

```
5 5
1 2 3
2 3 4
3 4 5
4 5 1
5 1 2
```

output

```
11
```