K. Odd trip plans

Problem ID: K

NTU is a country with n airports numbered through 1 to n. There are some airways, each of which connects two different airports bidirectionally. In other words, if an airway connects airports u and v, a passenger can move either from u to v or from v to u in a single flight. Airways may be newly established or abolished.

Mr. Oddytrip, who is a traveler loving odd numbers, plans a trip from an airport to another one by flights. Let's say that he boards k flights: A flight from airport p_1 to p_2, then from p_2 to p_3, then from p_3 to p_4, and so on, and finally from p_k to p_{k+1}. This trip plan, which begins with p_1 and ends with p_{k+1}, is written as $p_1 \to p_2 \to p_3 \to p_4 \to \cdots \to p_k \to p_{k+1}$. According to his aesthetics, a trip plan is *beautiful* if each of n airports appear an odd number of times in the trip plan. For example, if $n = 6$, trip plans $3 \to 4 \to 5 \to 6 \to 1 \to 2$ and $1 \to 2 \to 3 \to 4 \to 5 \to 3 \to 2 \to 3 \to 2 \to 6$ are beautiful while $1 \to 3 \to 6$ and $1 \to 2 \to 3 \to 4 \to 1 \to 5 \to 6$ aren't. In particular, each of the n airports appears at least one in a beautiful trip plan.

Initially, there are m airways. Then, you are given q queries, which should be processed in the order they are given. Each query is of one of the two kinds below:

- 1 x y: The existence of the airway between airports x and y changes. If there is already an airway between airports x and y, then such an airway is abolished. In other words, Mr. Oddytrip is no longer able to board the direct flight between airports x and y (until it is newly established again). On the other hand, if there wasn't such an airway before, an airway between airports x and y is newly established. In other words, Mr. Oddytrip can board a direct flight between airports x and y (until it is abolished again).
- 2 x y: You have to determine whether there can be a beautiful trip plan which begins with airport x and ends with airport y using the airways which are available at that time.

Input

The input consists of a single test case of the following format.

$n\ m\ q$
$u_1 v_1$
$\vdots$
$u_m\ v_m$
$t_1\ x_1\ y_1$
$\vdots$
$t_q\ x_q\ y_q$

The first line consists of three integers n, m and q, where n is the number of airports in NTU country, m is the number of airways which are initially available, and q is the number of queries.

The i-th of the following m lines consists two integers u_i and v_i representing that an airway between airports u_i and v_i is initially available.

The j-th of the following q lines consists of three integers t_j, x_j and y_j representing the type of the query and the numbers of two airports as described above.

- $2 \leq n, m, q \leq 10^5$
- $1 \leq u_i < v_i \leq n$
- It is guaranteed that all m airways are distinct
- $1 \leq t_j \leq 2$
- $1 \leq x_j < y_j \leq n$
- It is guaranteed that there is at least one query where $t_j = 2$

Output

For each query where $t_j = 2$, print "`Yes`" in a single line if there can be a beautiful trip plan which begins with airport x and ends with airport y. Otherwise, print "`No`" in a single line.

Sample Input 1

```
4 2 6
1 2
3 4
2 1 2
1 2 3
2 1 2
1 2 4
1 2 3
2 1 3
```

Sample Output 1

```
No
Yes
Yes
```

Sample Input 2

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

Sample Output 2

```
Yes
No
Yes
```

This page is intentionally left blank.