Union-find

Time limit: 2 sec.
Memory limit: 512MB

Description

Since we've learned union-find data structure, it's time to implement one! Initially, there will be n integers, from 1 to n, and all integers are separate from each other. So there will be n groups. You will be given m operations, in a chronological order. There are 2 types of operations, such as following.

- Type 1: Given two integers, x and y, merge the groups they belong to. If they already belonged to the same group, do nothing.
- Type 2: Given two integers, x and y, print whether they belong to the same group. Print "Yes" if they belong to the same group, and print "No" otherwise.

Input

The first line contains two integers, n and m. (1 ≤ n,m ≤100000)

The i-th line of the next q line consists of the following:

- The first integer of the line is t_i, the type of the i-th query. (t_i = 1 or 2).
- The next two integers of the line are x_i and y_i, the two integers. (1 <= x_i,y_i <= n, x_i≠y_i)

It is guaranteed that there is at least one query of type 2.

Output

For every query of type 2, print the appropriate string as stated on the description section, in a new line.

Sample I/O

Input(s)	Output(s)

5 7
1 1 2
1 2 3
1 1 4
2 2 4
2 2 5
1 4 5
2 2 5
	Yes
	No
	Yes