Dynamic Tree Connectivity

A forest of unrooted trees initially consists of N ($1 \le N \le 100{,}000$) single-vertex trees. The vertices are numbered from 1 to N.

Your task is to maintain that forest and answer connectivity queries.

All edges in the problem are **undirected**.

You will receive the following queries, where ($1 \le A, B \le N$) :

- **add** A B : add an edge between vertices A and B, where initially there is no path between A and B.
- **rem** A B : remove edge between vertices A and B, where initially there is an edge between A and B.
- **conn** A B : print **YES** if there is a path between A and B and **NO** otherwise, where A and B are different.

Input

The first line of input contains the number of initial single-vertex trees N and the number of queries M ($1 \le M \le 100{,}000$). The following M lines contain queries.

Output

For each **conn** query output **YES** or **NO**. Pay attention to letter case.

Example

Input:
5 11
conn 1 5
add 1 2
add 1 3
add 3 4
add 5 4
conn 1 5
rem 4 5
conn 1 5
rem 3 4
add 3 5
conn 1 5

Output:
NO
YES
NO
YES

This example will be the first test case.