

# A VCS Problem

## Problem ID: avcsproblem

In software engineering, a ‘version control system’ (VCS) is responsible for managing changes to a computer program, document or other information. In this problem, you need to implement a VCS for an undirected graph.

Initially, your graph has  $N$  vertices and 0 edges. The VCS needs to maintain several **snapshots** of the graph. These **snapshots** are indexed in chronological order by integers starting from 0. An initial **snapshot** of the graph is taken, whose index is 0. Then you need to do  $Q$  operations on the graph. Each operation is one of the following types:

- **ADD**  $u\ v$ : Add a new edge between two vertices  $u$  and  $v$ ;
- **CHECK**  $u\ v$ : In the current graph, check if two vertices  $u$  and  $v$  belong to the same connected component;
- **COMMIT** Create a new **snapshot** of the graph;
- **QUERY**  $s\ u\ v$  In the  $s$ -th **snapshot**, check if two vertices  $u$  and  $v$  belong to the same connected component.

### Input

The first line of the input contains two integers  $N$  and  $Q$  ( $1 \leq N, Q \leq 2 \times 10^5$ ). In the next  $Q$  lines, each describes an operation in any of the following forms:

- **A**  $u\ v$  ( $1 \leq u, v \leq N$ ) describes an **ADD** operation,
- **?**  $u\ v$  ( $1 \leq u, v \leq N$ ) describes a **CHECK** operation,
- **C** describes a **COMMIT** operation,
- **Q**  $s\ u\ v$  ( $1 \leq u, v \leq N, 0 \leq s < S$ ) where  $S$  is the number of snapshots taken so far, describes a **QUERY** operation.

### Output

For each **CHECK** or **QUERY** operation, print a character ‘Y’ if two vertices  $u$  and  $v$  belong to the same connected component, and print ‘N’ otherwise.

Sample Input 1	Sample Output 1
4 9 A 1 2 ? 1 2 Q 0 1 2 C Q 1 1 2 A 1 3 C Q 1 1 3 Q 2 1 3	YNYNY

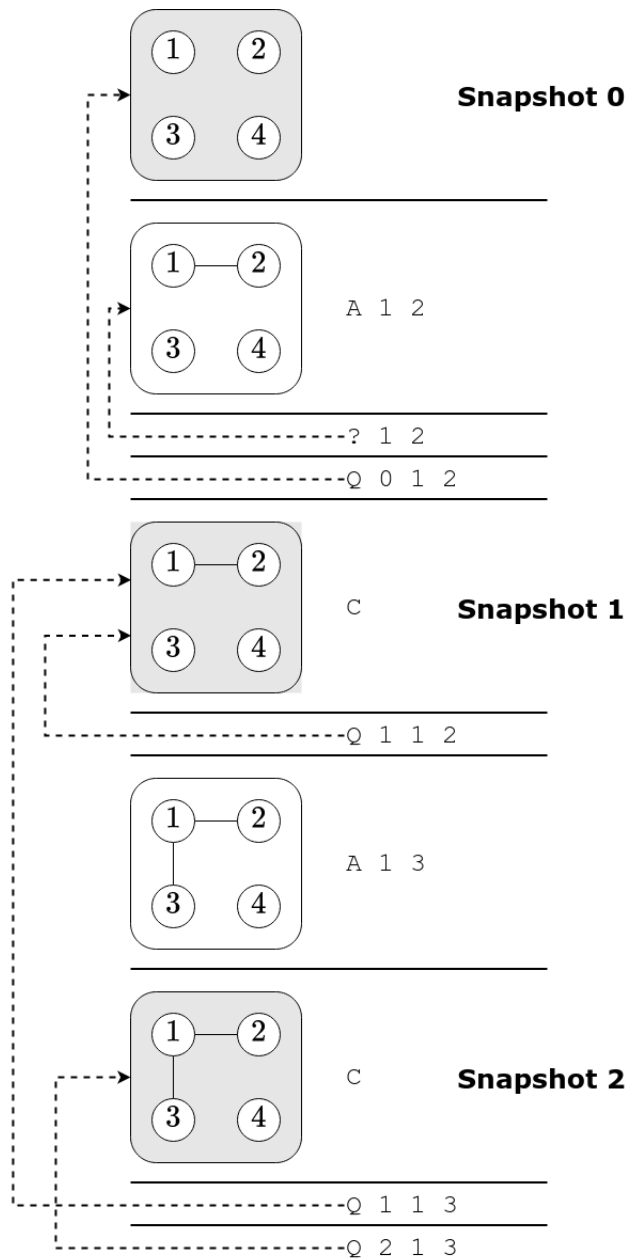


Figure 1: Explanation of the sample.

- The operations are executed from top to bottom.
- For each ? and Q operation, there is an arrow pointing to the corresponding snapshot of the graph.