I: **Impossible Communication**

Source file name: `impossible.c,` `impossible.cpp`, `impossible.java`, *or* `impossible.py`
Author: Rafael García

At your home university there are N research groups and M procedures to exchange information between them. Some procedures enable information to be sent from a specific group to another group. Other procedures are designed to share information between two or more groups (in either direction for each pair of them).

To be more precise, the research system of your university uses the following notation to identify the information sharing procedures:

- A *simple procedure* that enables information to be sent from the group I to the group J is specified as

$$1 \quad I \quad J$$

- A *complex procedure* that enables bi-directional information sharing between the k groups $I_1, I_2, \ldots, I_k$ is described as

$$k \quad I_1 \quad I_2 \quad \cdots \quad I_k$$

The president of your home university wants to make sure that the procedures to exchange information between research groups are *sufficiently complete*: he wants to make sure that any research group is able to send information to any other group (regardless of whether it is done directly or using intermediaries).

Please, help your president!

Input

The input consists of several test cases. The first line of a case contains two integers N and M indicating, respectively, the number of research groups ($2 \leq N \leq 50\,000$) and the number of information sharing procedures ($1 \leq M \leq 1\,000$). Each of the next M lines identifies one simple or one complex procedure, following the above-described notation. Research groups are represented with the integers $1, 2, \ldots, N$. You can assume that each complex procedure has at most $1\,000$ research groups. Single blanks are used to separate any pair of consecutive numbers.

The input must be read from standard input.

Output

For each test case output a single line with `YES` if the given set of procedures is sufficiently complete and `NO` otherwise.

The output must be written to standard output.

Sample Input	Sample Output
3 3 1 1 2 1 2 3 1 1 3 4 3 1 1 2 1 2 3 3 1 3 4	NO YES