

## Problem A. Alohomora and Colloportus

Source file name: Alohomora.c, Alohomora.cpp, Alohomora.java, Alohomora.py  
 Input: Standard  
 Output: Standard

*Colloportus* is a charm that magically locks objects such that they cannot be opened manually. On the other hand, *Alohomora*, also known as *Thief's Friend*, is an unlocking charm that can open even magically locked objects. Both charms are taught in Hogwarts charms class in the first year and, by the end of the fifth year, every student should be able to cast them, as they are required in the *Theory of Charms* examination.

In the exam, each student is given  $n$  interlocked chain links. To pass this part of the exam, you have to use the Alohomora charm to open some of the links and the Colloportus charm to interlock them again.

A messed up chain. Photo by Daniel Kirsch, Pixabay

By the end of the exam, the chain links should form a single closed chain. Ron messed up and is now left with a bunch of partially interlocked chain links and only enough time to cast both spells a single time.

More formally, Ron is able to open one chain link, change with which other chain links it is interlocked with, and then lock it again. After this, each chain link should be interlocked with exactly two other chain links to form a single closed chain. Is it still possible for Ron to pass the exam?

### Input

The input consists of:

- One line with two integers  $n$  and  $m$  ( $3 \leq n \leq 10^5, 0 \leq m \leq 2 \cdot 10^5$ ), the number of chain links and the number of interlocked pairs of chain links.
- $m$  lines, each containing two integers  $a$  and  $b$  ( $1 \leq a < b \leq n$ ), which means that the  $a$ -th and  $b$ -th chain link are interlocked.

Note that each pair of interlocked chain links is given at most once and that the initial chain may not be connected.

### Output

If Ron can form a closed chain, output **yes**. Otherwise, output **no**.

### Example

Input	Output
4 3 1 2 2 3 2 4	yes
4 6 1 2 1 3 1 4 2 3 2 4 3 4	no