Summer Institute '17 Contest 3 P4 - Customs

Time limit: 1.4s   **Memory limit:** 256M
C: 0.75s
C++: 0.75s

Summer Institute @ University of Central Florida: Contest 3, Problem 4

Gennady loves to travel; one might even call him a professional tourist. On most of his trips, he will bring graphs with him to play with. Unfortunately for him, some of his graphs are cacti. He is worried that US customs will classify his cacti as a prohibited agricultural item and prevent him from bringing them into the country.

So he needs to identify which of his graphs are cacti. Sometimes the US will allow cacti into the country if they meet certain spanning tree limitations. Namely, the US requires that the number of spanning trees that can be formed from the given cactus does not exceed a certain amount. Therefore, he would like you to check if a given graph in his collection is a cactus and, if so, report the number of spanning trees of the cactus.

A cactus is a connected graph where each edge is involved in at most one cycle. A spanning tree is a subgraph of a given graph that is both a tree and every vertex of the original graph is included in the tree. Two spanning trees are considered different if they differ in the edges they include.

Input Specification

The first line contains two integers n and m ($1 \leq n \leq 2 \times 10^4, 1 \leq m \leq 10^5$), the number of nodes and number of edges in the graph, respectively.

The next m lines contain two integers, a_i and b_i ($1 \leq a_i \neq b_i \leq n$), representing an edge between nodes a_i and b_i.

Output Specification

If the graph given is not a cactus, output `safe`. Otherwise output a single integer, the number of spanning trees of the given cactus. As this number can be quite large, output the result modulo $10^9 + 7$.

Sample Input 1

```
14 15
1 2
2 3
3 4
4 5
5 6
6 7
7 8
8 3
2 9
9 10
10 11
11 12
12 13
13 10
2 14
```

Sample Output 1

```
24
```

Sample Input 2

```
10 11
1 2
2 3
3 4
4 5
5 6
6 1
3 7
7 8
8 9
9 10
10 2
```

Sample Output 2

```
safe
```

Sample Input 3

```
3 1
1 3
```

Sample Output 3

```
safe
```