Blueprint Feasibility

Problem Description

Alex works at a smart factory that assembles products from a customer's blueprint. The assembly process is broken down into many tasks, and some tasks must be completed before others—in other words, there are dependencies between tasks.

Given these dependencies, please help Alex determine whether the entire product can be assembled without deadlock.

Input Format

The first line contains two integers ND:

- N — number of tasks, labeled $0, 1, ..., N-1$
- D — number of dependencies between two different tasks

The next D lines each has two integers u v meaning task u must be done before task v.

Output Format

One line with a single integer:

- 0 if the product can be assembled.
- 1 if the product can never be assembled.

Constraints

- $1 \leq N \leq 2 \times 10^6$ ∘
- $0 \leq D \leq 2.7 \times 10^6$ ∘
- $0 \leq u, v \leq N-1$ ∘
- No duplicate dependencies (no repeated u v pairs).
- No self-dependency ($\forall$ dependency pair u $v, u \neq v$).

Example Test Case

Sample Input 1

```
3 3
0 1
1 2
2 0
```

Sample Output 1

```
1
```

Explanation of Sample 1

The dependencies $0 \to 1, 1 \to 2$, and $2 \to 0$ create a directed cycle. Therefore, no topological ordering exists and the plan is infeasible.

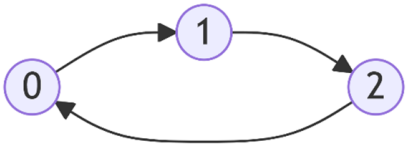

Figure 1: Graph of Sample Input 1

Sample Input 2

```
3 2
0 1
2 1
```

Sample Output 2

```
0
```

Explanation of Sample 2

The product can be assembled in either order $0 \to 2 \to 1$ *or* $2 \to 0 \to 1$.

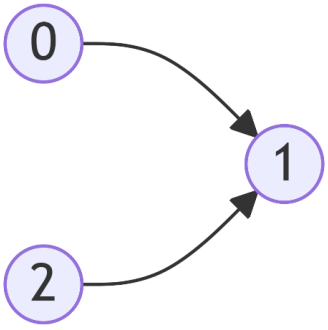

Figure 2: Graph of Sample Input 2

Sample Input 3

```
4 3
0 1
2 3
0 3
```

Sample Output 3

```
0
```

Explanation of Sample 3

The product can be assembled. For example, in the order $0 \to 1 \to 2 \to 3$.

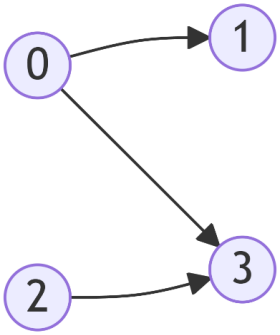

Figure 3: Graph of Sample Input 3