In a town, there are `n` people labeled from `1` to `n`. There is a rumor that one of these people is secretly the town judge.

If the town judge exists, then:

1. The town judge trusts nobody.
2. Everybody (except for the town judge) trusts the town judge.
3. There is exactly one person that satisfies properties **1** and **2**.

You are given an array `trust` where `trust[i] = [a_i, b_i]` representing that the person labeled `a_i` trusts the person labeled `b_i`.

Return *the label of the town judge if the town judge exists and can be identified, or return* `-1` *otherwise.*

Example 1:

```
Input: n = 2, trust = [[1,2]]
Output: 2
```

Example 2:

```
Input: n = 3, trust = [[1,3],[2,3]]
Output: 3
```

Example 3:

```
Input: n = 3, trust = [[1,3],[2,3],[3,1]]
Output: -1
```

Constraints:

- `1 <= n <= 1000`
- `0 <= trust.length <= 10^4`
- `trust[i].length == 2`
- All the pairs of `trust` are **unique**.
- `a_i != b_i`
- `1 <= a_i, b_i <= n`