997. Find the Town Judge

Solved

Easy | Topics | Companies

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`. If a trust relationship does not exist in `trust` array, then such a trust relationship does not exist.

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

```
class Solution {
    func findJudge(_ n: Int, _ trust: [[Int]]) -> Int {
        if(n == 1) {
            return 1
        }
        var arr = Array(repeating:0, count:n+1)
        var hash:[Int:Int] = [:]
        for i in trust {
            arr[i[1]] = arr[i[1]] + 1
            hash[i[0]] = i[1]
        }

        // print(arr)
        // print(hash)
        var ans = -1
        for i in (0..<arr.count) {
            if(arr[i] == n-1) {
                if let v = hash[i] {
```

```
                    return -1
                } else {
                    return i
                }
            }
        }

        return ans
    }
}
```