

2385. Amount of Time for Binary Tree to Be Infected

Solved ✓

Medium

Topics

Companies

Hint

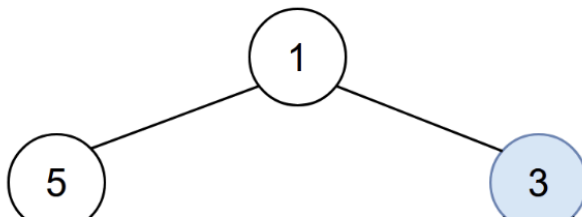
You are given the `root` of a binary tree with **unique** values, and an integer `start`. At minute `0`, an **infection** starts from the node with value `start`.

Each minute, a node becomes infected if:

- The node is currently uninfected.
- The node is adjacent to an infected node.

Return *the number of minutes needed for the entire tree to be infected*.

Example 1:



```
/**
 * Definition for a binary tree node.
 * public class TreeNode {
 *     public var val: Int
 *     public var left: TreeNode?
 *     public var right: TreeNode?
 *     public init() { self.val = 0; self.left = nil; self.right =
nil; }
 *     public init(_ val: Int) { self.val = val; self.left = nil;
self.right = nil; }
 *     public init(_ val: Int, _ left: TreeNode?, _ right:
TreeNode?) {
 *         self.val = val
 *         self.left = left
 *         self.right = right

```

```

*      }
* }
*/

class Solution {
    func amountOfTime(_ root: TreeNode?, _ start: Int) -> Int {

        if(root!.left == nil && root!.right == nil) {
            return 0
        }

        var graph:[Int:[Int]] = [:]
        convertTreeToGraph(root,&graph)
        var count = 0
        var visited = Set<Int>()
        var queue = [Int]()
        queue.append(start)
        visited.insert(start)

        while (!queue.isEmpty ) {
            print("one iteration ")
            let q = queue.count
            for i in 0..

```

```
func convertTreeToGraph(_ root: TreeNode?, _ graph: inout
[Int:[Int]]) {
    guard let root = root else {
        return
    }

    if(root.left != nil) {
        graph[root.val, default:[]].append(root.left!.val)
        graph[root.left!.val, default:[]].append(root.val)
    }
    if(root.right != nil) {
        graph[root.val, default:[]].append(root.right!.val)
        graph[root.right!.val, default:[]].append(root.val)
    }

    convertTreeToGraph(root.left, &graph)
    convertTreeToGraph(root.right, &graph)
}
```