

# Problem 3313: Find the Last Marked Nodes in Tree

## Problem Information

Difficulty: **Hard**

Acceptance Rate: 53.71%

Paid Only: Yes

Tags: Tree, Depth-First Search

## Problem Description

There exists an **undirected** tree with  $n$  nodes numbered  $0$  to  $n - 1$ . You are given a 2D integer array `edges` of length  $n - 1$ , where `edges[i] = [ui, vi]` indicates that there is an edge between nodes `ui` and `vi` in the tree.

Initially, **all** nodes are **unmarked**. After every second, you mark all unmarked nodes which have **at least** one marked node `_adjacent_` to them.

Return an array `nodes` where `nodes[i]` is the last node to get marked in the tree, if you mark node `i` at time `t = 0`. If `nodes[i]` has `_multiple_` answers for any node `i`, you can choose **any** one answer.

**Example 1:**

**Input:** `edges = [[0,1],[0,2]]`

**Output:** `[2,2,1]`

**Explanation:**



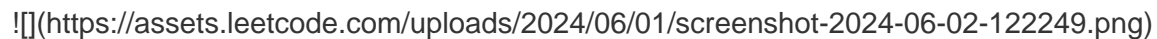
\* For `i = 0`, the nodes are marked in the sequence: `[0] -> [0,1,2]`. Either 1 or 2 can be the answer. \* For `i = 1`, the nodes are marked in the sequence: `[1] -> [0,1] -> [0,1,2]`. Node 2 is marked last. \* For `i = 2`, the nodes are marked in the sequence: `[2] -> [0,2] -> [0,1,2]`. Node 1 is marked last.

**Example 2.**

**Input:** edges = [[0,1]]

**Output:** [1,0]

**Explanation:**

 (https://assets.leetcode.com/uploads/2024/06/01/screenshot-2024-06-02-122249.png)

\* For  $i = 0$ , the nodes are marked in the sequence:  $[0] \rightarrow [0,1]$ . \* For  $i = 1$ , the nodes are marked in the sequence:  $[1] \rightarrow [0,1]$ .

**Example 3.**

**Input:** edges = [[0,1],[0,2],[2,3],[2,4]]

**Output:** [3,3,1,1,1]

**Explanation:**

 (https://assets.leetcode.com/uploads/2024/06/03/screenshot-2024-06-03-210550.png)

\* For  $i = 0$ , the nodes are marked in the sequence:  $[0] \rightarrow [0,1,2] \rightarrow [0,1,2,3,4]$ . \* For  $i = 1$ , the nodes are marked in the sequence:  $[1] \rightarrow [0,1] \rightarrow [0,1,2] \rightarrow [0,1,2,3,4]$ . \* For  $i = 2$ , the nodes are marked in the sequence:  $[2] \rightarrow [0,2,3,4] \rightarrow [0,1,2,3,4]$ . \* For  $i = 3$ , the nodes are marked in the sequence:  $[3] \rightarrow [2,3] \rightarrow [0,2,3,4] \rightarrow [0,1,2,3,4]$ . \* For  $i = 4$ , the nodes are marked in the sequence:  $[4] \rightarrow [2,4] \rightarrow [0,2,3,4] \rightarrow [0,1,2,3,4]$ .

**Constraints:**

\*  $2 \leq n \leq 105$  \*  $\text{edges.length} == n - 1$  \*  $\text{edges}[i].\text{length} == 2$  \*  $0 \leq \text{edges}[i][0], \text{edges}[i][1] \leq n - 1$  \* The input is generated such that `edges` represents a valid tree.

## Code Snippets

**C++:**

```
class Solution {  
public:  
    vector<int> lastMarkedNodes(vector<vector<int>>& edges) {  
  
    }  
};
```

### Java:

```
class Solution {  
    public int[] lastMarkedNodes(int[][] edges) {  
  
    }  
}
```

### Python3:

```
class Solution:  
    def lastMarkedNodes(self, edges: List[List[int]]) -> List[int]:
```