AlgoExpert   Quad Layout   Swift   12px   Sublime   Monokai   00:00:00

Prompt   Scratchpad   Our Solution(s)   Video Explanation   Run Code

Solution 1   Solution 2

```swift
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

class Program {
    class BinaryTree {
        var value: Int
        var left: BinaryTree?
        var right: BinaryTree?

        init(value: Int) {
            self.value = value
            left = nil
            right = nil
        }
    }

    // O(n) time | O(d) space - where n is the number of nodes in the Binary Tree
    // and d is the depth (height) of the Binary Tree
    func flattenBinaryTree(root: BinaryTree) -> BinaryTree {
        var result = flattenTree(node: root)
        return result.leftMost
    }

    func flattenTree(node: BinaryTree) -> (leftMost: BinaryTree, rightMost: BinaryTree) {
        var leftMost = node
        if let left = node.left {
            var result = flattenTree(node: left)
            connectNodes(left: result.rightMost, right: node)
            leftMost = result.leftMost
        }

        var rightMost = node
        if let right = node.right {
            var result = flattenTree(node: right)
            connectNodes(left: node, right: result.leftMost)
            rightMost = result.rightMost
        }
        return (leftMost, rightMost)
    }

    func connectNodes(left: BinaryTree, right: BinaryTree) {
        left.right = right
        right.left = left
    }
}
```