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

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1 Solution 2

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

package main

type BinaryTree struct {
  Value int

  Left  *BinaryTree
  Right *BinaryTree
}

// 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 {
  leftMost, _ := flattenTree(root)
  return leftMost
}

func flattenTree(node *BinaryTree) (leftMost, rightMost *BinaryTree) {
  leftMost = node
  if node.Left != nil {
    leftSubtreeLeftMost, leftSubtreeRightMost := flattenTree(node.Left)
    connectNodes(leftSubtreeRightMost, node)
    leftMost = leftSubtreeLeftMost
  }

  rightMost = node
  if node.Right != nil {
    rightSubtreeLeftMost, rightSubtreeRightMost := flattenTree(node.Right)
    connectNodes(node, rightSubtreeLeftMost)
    rightMost = rightSubtreeRightMost
  }
  return leftMost, rightMost
}

func connectNodes(left, right *BinaryTree) {
  left.Right = right
  right.Left = left
}
```