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

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1

```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 RightSiblingTree(root *BinaryTree) *BinaryTree {
  mutate(root, nil, false)
  return root
}

func mutate(node, parent *BinaryTree, isLeftChild bool) {
  if node == nil {
    return
  }

  left, right := node.Left, node.Right
  mutate(left, node, true)
  if parent == nil {
    node.Right = nil
  } else if isLeftChild {
    node.Right = parent.Right
  } else {
    if parent.Right == nil {
      node.Right = nil
    } else {
      node.Right = parent.Right.Left
    }
  }
  mutate(right, node, false)
}
```