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

Run Code Scratchpad Our Solution(s) Video Explanation

```
Solution 1
```

Prompt

41

```
_{\rm 1} \, // Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
 3 public class Program {
     // O(n) time \mid O(d) space - where n is the number of nodes in
      // the Binary Tree and d is the depth (height) of the Binary Tree
      public static BinaryTree RightSiblingTree(BinaryTree root) {
        mutate(root, null, false);
        return root;
 9
10
      public static void mutate(BinaryTree node, BinaryTree parent, bool isLeftChild) {
11
12
        if (node == null) return;
13
14
        var left = node.left;
        var right = node.right;
mutate(left, node, true);
15
16
17
        if (parent == null) {
18
          node.right = null;
        } else if (isLeftChild) {
19
20
21
          node.right = parent.right;
        } else{
22
          if (parent.right == null) {
23
            node.right = null;
24
          } else {
25
            node.right = parent.right.left;
26
27
28
        mutate(right, node, false);
29
30
31
      public class BinaryTree {
32
        public int value;
33
        public BinaryTree left = null;
34
        public BinaryTree right = null;
35
36
        public BinaryTree(int value) {
37
          this.value = value;
38
39
40 }
```