```
1  // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3  public class Program {
4    // Average case: when the tree is balanced
5    // O(n) time | O(h) space - where n is the number of nodes in
6    // the Binary Tree and h is the height of the Binary Tree
7    public static int NodeDepths(BinaryTree root) {
8      return nodeDepthsHelper(root, 0);
9    }
10
11   public static int nodeDepthsHelper(BinaryTree root, int depth) {
12     if (root == null) return 0;
13     return depth + nodeDepthsHelper(root.left, depth + 1) + nodeDepthsHelper(root.right,
14              depth + 1);
15   }
16
17   public class BinaryTree {
18     public int value;
19     public BinaryTree left;
20     public BinaryTree right;
21
22     public BinaryTree(int value) {
23       this.value = value;
24       left = null;
25       right = null;
26     }
27   }
28 }
29
```