AlgoExpert Quad Layout Go 12px Sublime Monok

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1 Solution 2 Solution 3 Solution 4

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

package main

type BinaryTree struct {
  Value       int
  Left, Right *BinaryTree
}

// Average case: when the tree is balanced
// O(nlog(n)) time | O(h) space - where n is the number of nodes in
// the Binary Tree and h is the height of the Binary Tree
func AllKindsOfNodeDepths(root *BinaryTree) int {
  sumOfDepths := 0
  stack := []*BinaryTree{root}
  var node *BinaryTree
  for len(stack) > 0 {
    node, stack = stack[len(stack)-1], stack[:len(stack)-1]
    if node == nil {
      continue
    }
    sumOfDepths += nodeDepths(node, 0)
    stack = append(stack, node.Left)
    stack = append(stack, node.Right)
  }
  return sumOfDepths
}

func nodeDepths(node *BinaryTree, depth int) int {
  if node == nil {
    return 0
  }
  return depth + nodeDepths(node.Left, depth+1) + nodeDepths(node.Right, depth+1)
}
```