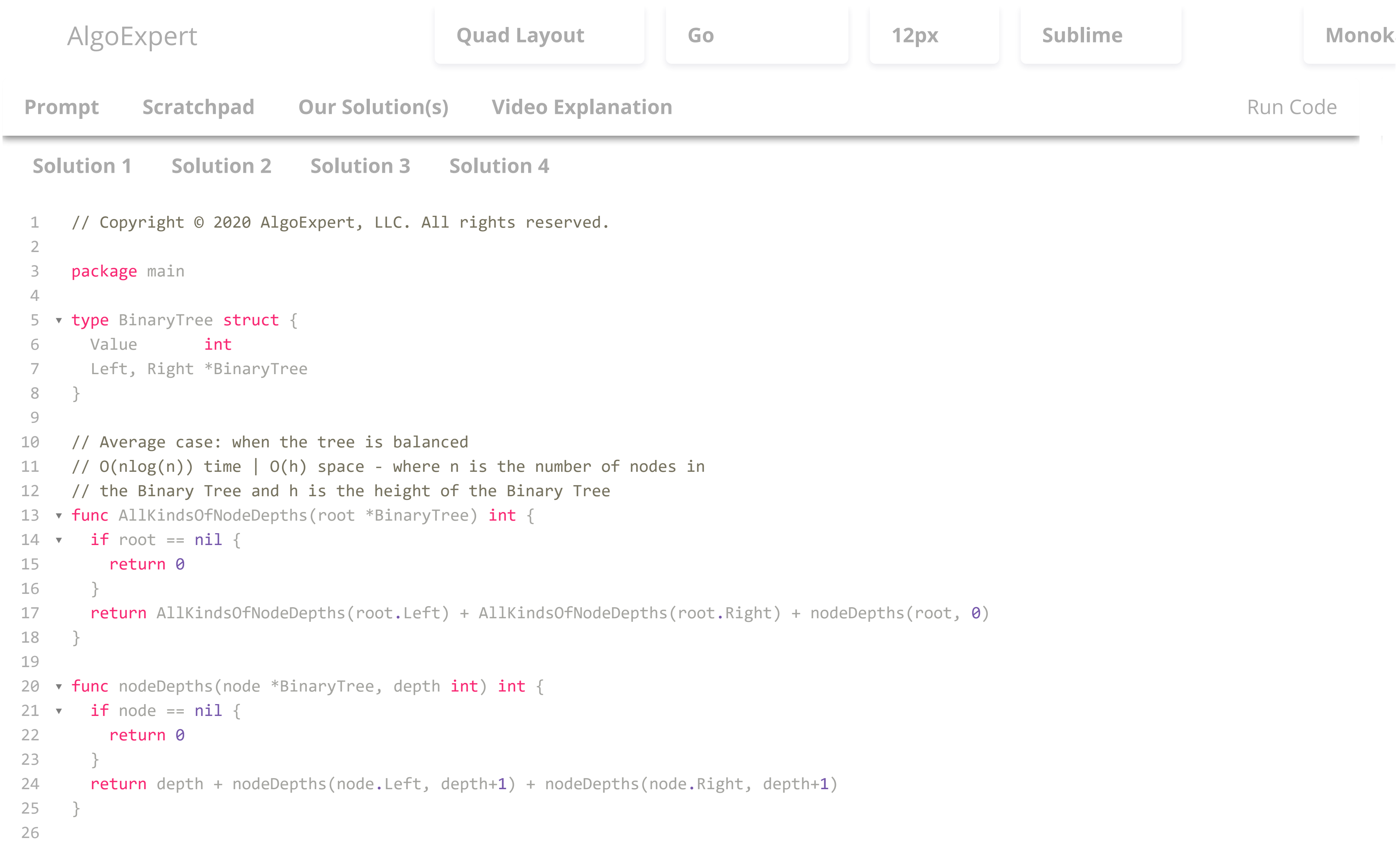

```go
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

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 {
  if root == nil {
    return 0
  }
  return AllKindsOfNodeDepths(root.Left) + AllKindsOfNodeDepths(root.Right) + nodeDepths(root, 0)
}

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