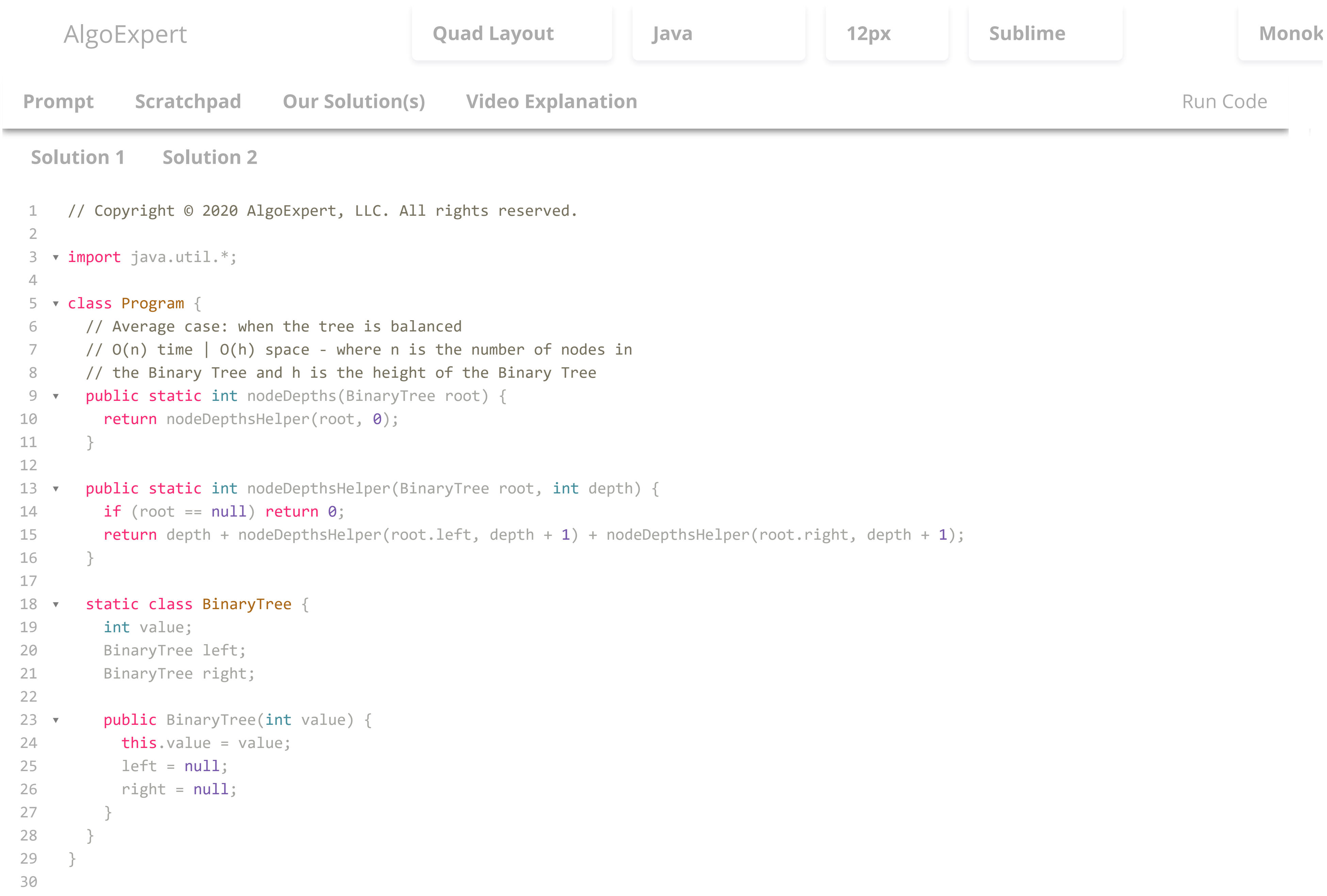

```java
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

import java.util.*;

class Program {
  // Average case: when the tree is balanced
  // O(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
  public static int nodeDepths(BinaryTree root) {
    return nodeDepthsHelper(root, 0);
  }

  public static int nodeDepthsHelper(BinaryTree root, int depth) {
    if (root == null) return 0;
    return depth + nodeDepthsHelper(root.left, depth + 1) + nodeDepthsHelper(root.right, depth + 1);
  }

  static class BinaryTree {
    int value;
    BinaryTree left;
    BinaryTree right;

    public BinaryTree(int value) {
      this.value = value;
      left = null;
      right = null;
    }
  }
}

```