

PromptScratchpadOur Solution(s)Video Explanation

Run Code

Solution 1

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 using System;
4 using System.Collections.Generic;
5
6 public class Program {
7     // O(n) time | O(log(n)) space
8     public static int MaxPathSum(BinaryTree tree) {
9         List<int> maxSumArray = findMaxSum(tree);
10        return maxSumArray[1];
11    }
12
13    public static List<int> findMaxSum(BinaryTree tree) {
14        if (tree == null) {
15            return new List<int>(){
16                0, 0
17            };
18        }
19        List<int> leftMaxSumArray = findMaxSum(tree.left);
20        int leftMaxSumAsBranch = leftMaxSumArray[0];
21        int leftMaxPathSum = leftMaxSumArray[1];
22
23        List<int> rightMaxSumArray = findMaxSum(tree.right);
24        int rightMaxSumAsBranch = rightMaxSumArray[0];
25        int rightMaxPathSum = rightMaxSumArray[1];
26
27        int maxChildSumAsBranch = Math.Max(leftMaxSumAsBranch, rightMaxSumAsBranch);
28        int maxSumAsBranch = Math.Max(maxChildSumAsBranch + tree.value, tree.value);
29        int maxSumAsRootNode = Math.Max(
30            leftMaxSumAsBranch + tree.value + rightMaxSumAsBranch,
31            maxSumAsBranch
32        );
33        int maxPathSum = Math.Max(leftMaxPathSum, Math.Max(rightMaxPathSum,
34            maxSumAsRootNode));
35
36        return new List<int>(){
37            maxSumAsBranch, maxPathSum
38        };
39    }
40
41    public class BinaryTree {
42        public int value;
43        public BinaryTree left;
44        public BinaryTree right;
45
46        public BinaryTree(int value) {
47            this.value = value;
48        }
49    }
50 }
51
```