AlgoExpert   Quad Layout   C#   14px   Sublime   Monokai   00:00:0

Our Solution(s)   Run Code

Solution 1

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

using System.Collections.Generic;

public class Program {
  public class BinaryTree {
    public int value;
    public BinaryTree left;
    public BinaryTree right;

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

  // O(n) time | O(n) space - where n is the number of nodes in the Bi
  public static List<int> BranchSums(BinaryTree root) {
    List<int> sums = new List<int>();
    calculateBranchSums(root, 0, sums);
    return sums;
  }

  public static void calculateBranchSums(BinaryTree node, int runningS
    if (node == null) return;

    int newRunningSum = runningSum + node.value;
    if (node.left == null && node.right == null) {
      sums.Add(newRunningSum);
      return;
    }

```

Your Solutions   Run Code

Solution 1   Solution 2   Solution 3

```
using System.Collections.Generic;

public class Program {
  // This is the class of the input root. Do not edit it.
  public class BinaryTree {
    public int value;
    public BinaryTree left;
    public BinaryTree right;

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

  public static List<int> BranchSums(BinaryTree root) {
    // Write your code here.
    return null;
  }
}

```

Our Tests

[illegible]

Custom Output   Submit Code

[illegible]

Run or submit code when you're ready.