AlgoExpert Quad Layout C++ 14px Sublime Monokai 00:00:

Our Solution(s) Run Code

Solution 1

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

#include <vector>

using namespace std;

class BinaryTree {
public:
  int value;
  BinaryTree *left;
  BinaryTree *right;

  BinaryTree(int value) {
    this->value = value;
    left = NULL;
    right = NULL;
  }
};

void calculateBranchSums(BinaryTree *node, int runningSum, vector<int>

// O(n) time | O(n) space - where n is the number of nodes in the Bina
vector<int> branchSums(BinaryTree *root) {
  vector<int> sums;
  calculateBranchSums(root, 0, sums);
  return sums;
}

void calculateBranchSums(BinaryTree *node, int runningSum, vector<int>
  if (node == NULL)
    return;

  int newRunningSum = runningSum + node->value;
```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```cpp
using namespace std;

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

  BinaryTree(int value) {
    this->value = value;
    left = NULL;
    right = NULL;
  }
};

vector<int> branchSums(BinaryTree *root) {
  // Write your code here.
  return {};
}
```

Our Tests

[illegible]

Custom Output Submit Code

```
[illegible]
```

Run or submit code when you're ready.