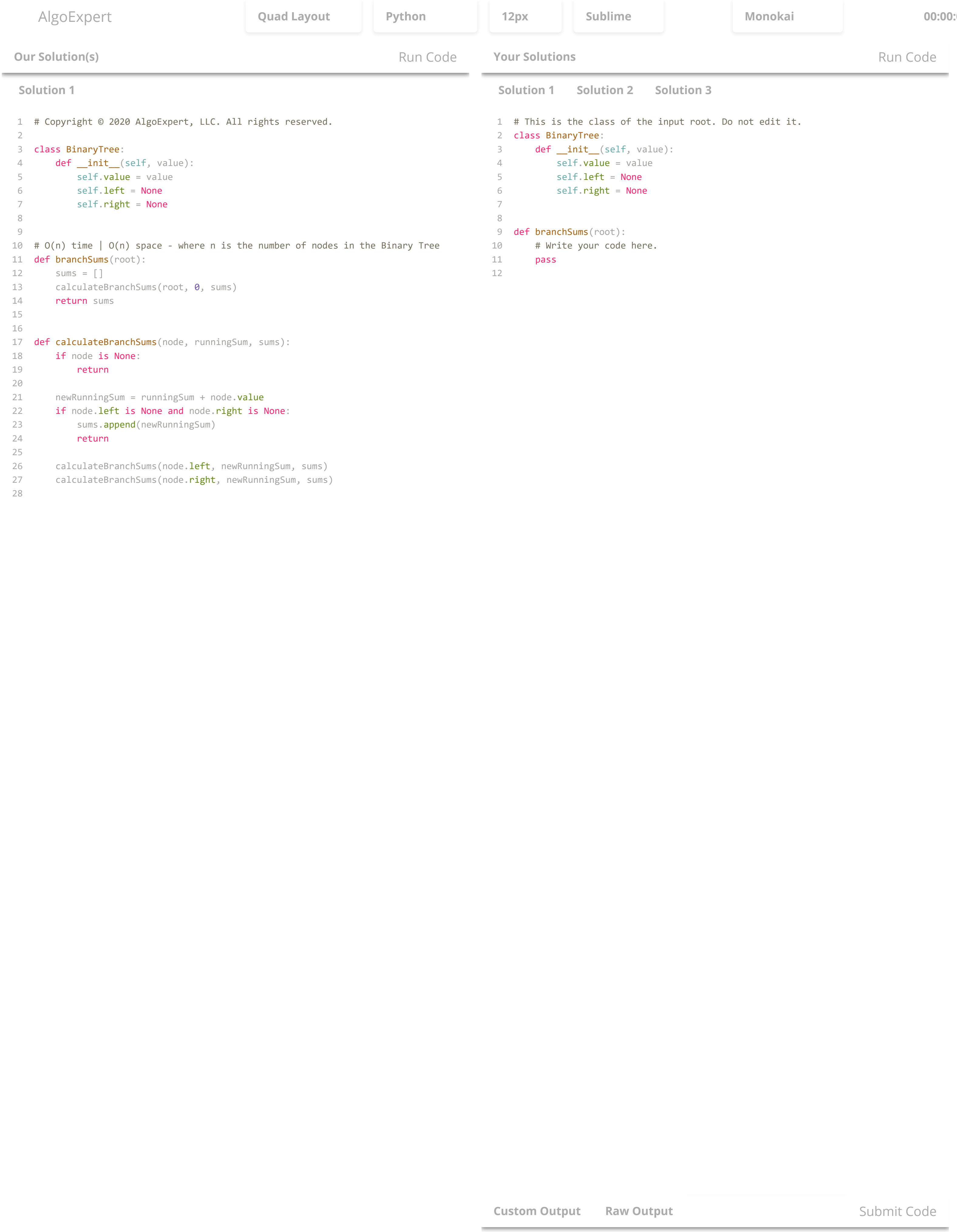

```python
# Copyright © 2020 AlgoExpert, LLC. All rights reserved.

class BinaryTree:
    def __init__(self, value):
        self.value = value
        self.left = None
        self.right = None


# O(n) time | O(n) space - where n is the number of nodes in the Binary Tree
def branchSums(root):
    sums = []
    calculateBranchSums(root, 0, sums)
    return sums


def calculateBranchSums(node, runningSum, sums):
    if node is None:
        return

    newRunningSum = runningSum + node.value
    if node.left is None and node.right is None:
        sums.append(newRunningSum)
        return

    calculateBranchSums(node.left, newRunningSum, sums)
    calculateBranchSums(node.right, newRunningSum, sums)

```

```python
# This is the class of the input root. Do not edit it.
class BinaryTree:
    def __init__(self, value):
        self.value = value
        self.left = None
        self.right = None


def branchSums(root):
    # Write your code here.
    pass

```

Run or submit code when you're ready.

Our Tests