Our Solution(s) Run Code

Solution 1

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

# O(n) time | O(n) space
def inOrderTraverse(tree, array):
    if tree is not None:
        inOrderTraverse(tree.left, array)
        array.append(tree.value)
        inOrderTraverse(tree.right, array)
    return array


# O(n) time | O(n) space
def preOrderTraverse(tree, array):
    if tree is not None:
        array.append(tree.value)
        preOrderTraverse(tree.left, array)
        preOrderTraverse(tree.right, array)
    return array


# O(n) time | O(n) space
def postOrderTraverse(tree, array):
    if tree is not None:
        postOrderTraverse(tree.left, array)
        postOrderTraverse(tree.right, array)
        array.append(tree.value)
    return array

```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```python
def inOrderTraverse(tree, array):
    # Write your code here.
    pass


def preOrderTraverse(tree, array):
    # Write your code here.
    pass


def postOrderTraverse(tree, array):
    # Write your code here.
    pass

```

Our Tests

Custom Output Submit Code

```
import unittest


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

    def insert(self, value):
        if value < self.value:
            if self.left is None:
                self.left = BST(value)
            else:
                self.left.insert(value)
        else:
            if self.right is None:
                self.right = BST(value)
            else:
                self.right.insert(value)
        return self
```

Run or submit code when you're ready.