NAME ____ **ICA-13**

NetID ____________________@email.arizona.edu

NOTES: This a group activity. **Speak** to your neighbor! This ICA will be graded for participation.

---

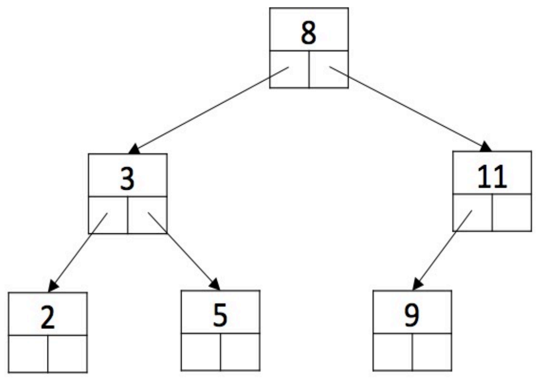
1. Consider the definition below of a BinarySearchTree:

```
class Node:
    def __init__(self, value):
        self._value = value
        self._lchild = None
        self._rchild = None

class BinarySearchTree:
    def __init__(self, value):
        self._value = value
        self._lchild = None
        self._rchild = None
```

   a) Write `insert(node)` function in BinarySearchTree to insert a node into the tree, and maintain it as a binary search tree. This function returns the inserted node. If a node in the tree has the value, return the node.
   The following code can create a tree shown in next page:

```
BST = BinarySearchTree(None)
BST.insert(2)
BST.insert(3)
BST.insert(5)
BST.insert(8)
BST.insert(9)
BST.insert(11)
```



b) Write a function `traverse(t)` for a BST that prints the values in the tree, one value per line, in this order: 2, 3, 5, 8, 9, 11. **Use a recursive solution!**

For convenience, you may directly access the attributes of the `BinarySearchTree` class when writing your function.

2. Given the definition of a `BinarySearchTree`, write a function `count_leaves(t)` that counts the number of leaf nodes.

3. Draw 2 trees from the following in-order traversal: 1 2 3 4