Homework 07 (Tree) (10 marks)

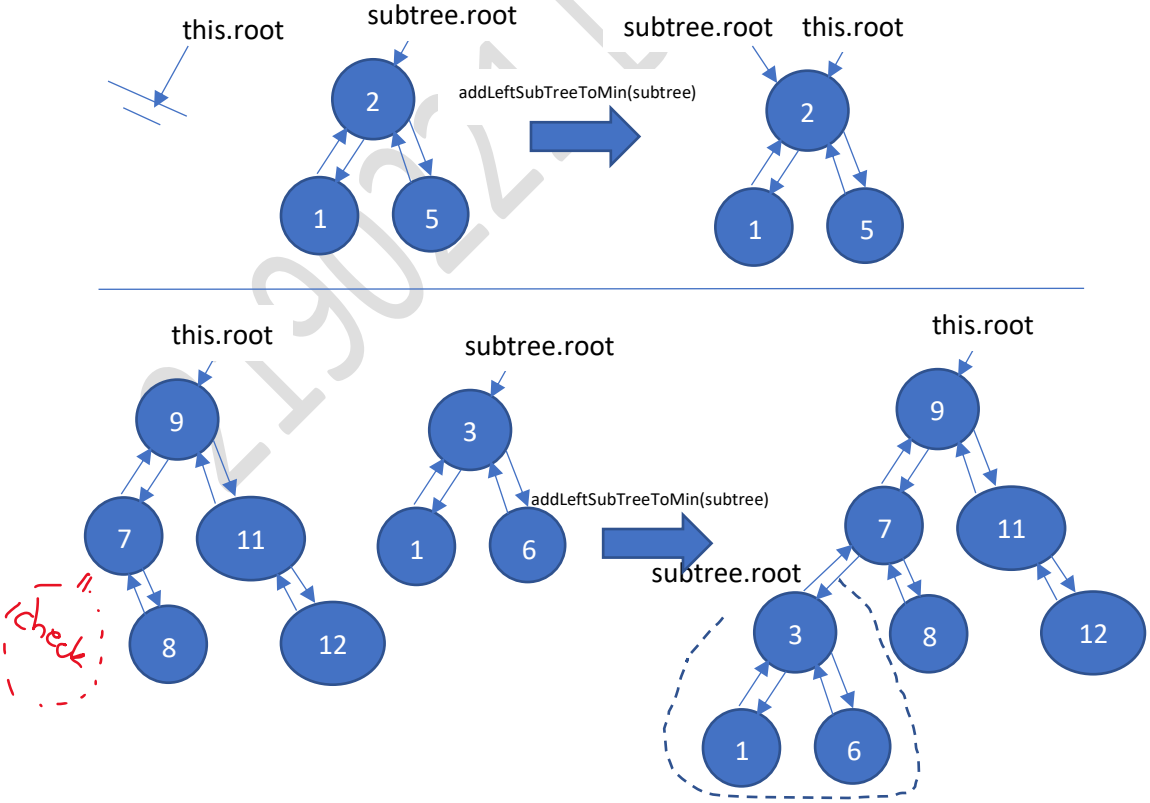
You are given all classes for coding a binary search tree.

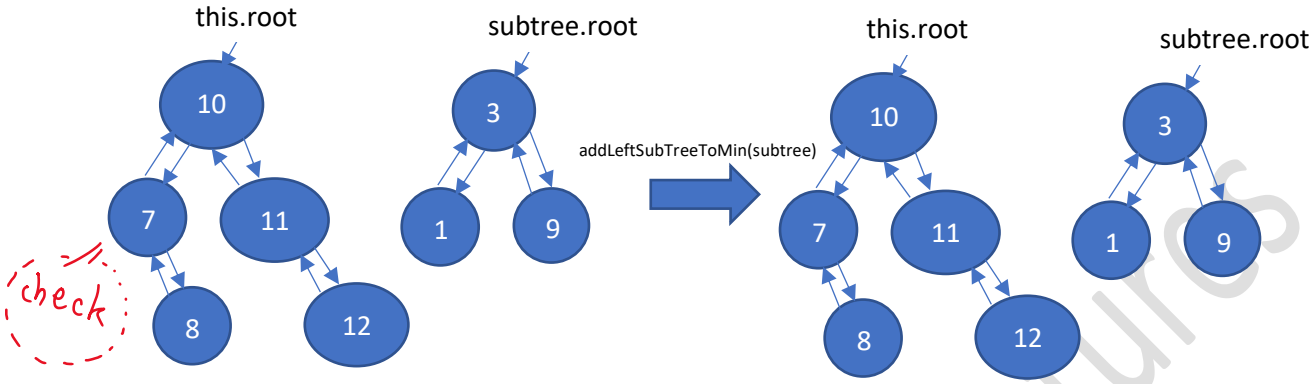
In class BST, write method

```
public void addLeftSubTreeToMin(BST subtree)
```

- This method tries to add an entire "subtree" as a left subtree of the left most node in our tree, changing our tree.
- Assume there will be no direct access, from anywhere else apart from our BST, to "subtree" in the future.
- If the "subtree" is an empty tree, this method does nothing.
- If our tree is empty, then our tree becomes the subtree.
- Before doing change, check whether after the addition of "subtree", the tree will still be a binary search tree:
  - If so, link the entire subtree to our tree.
  - If not, do nothing.
- The method must not have any loop (but you can call existing methods that have loop).
- Only BST.java is allowed to be modified.
  - Only modify this method. You are not allowed to create new method(s).
- This method must be the last method in class BST (not counting main method).
- Submit only BST.java on Mycourseville.

Example:





Nothing changes because we won't get a binary search tree.

The JUnit tests are in BSTTest.java (If you don't do any proper coding you won't get any mark)

- `testAddEmptySubTree()` 1 marks
- `testAddToEmptyTree()` 2 marks
- `testAddSuccess()` 3 marks
- `testAddFail()` 2 marks
- `testNoLoop()` 2 marks