Homework 6 – Due Apr. 24th 23:59 KST

Instructions: Complete the implementation and turn it in before the due date. Any deviations from the instructed deliverable format will result in a deduction of grade. DO NOT COPY OTHER'S WORKS!

In this assignment, you will implement various tree-related operations, some recursively and some non-recursively. This tree only contains unique elements, so any duplicate insertions should fail. There are seven required methods, from switchLR() through size(). Carefully read all the comments provided in the source file (especially the insertAt() since it's quite complicated).

Restrictions & conditions: No 'trivial' recursion, where the core work is done iteratively but a non-important side information is computed recursively, and no usage of techniques/methods not taught in class. The last restriction is to ensure prevention of unnecessary misunderstandings regarding cheating. You are, however, allowed to have helper methods that do the main recursion.

Rubric: Grading will be based on, but not limited to, the following criteria.

- Documentation (30 points): For each of the required main methods, you should either provide time complexity analysis (for non-recursive methods) or base- and recursive-case descriptions (for recursive methods).
- Correctness (70 points): Your implementation should behave as specified in the comments in an error-free manner. Two or more uncaught exceptions will result in a 0 for correctness.

Deliverable: A single BinaryTree.java file not part of any package structures.