Hands-On: AVL Trees

This activity focuses on the fundamental mechanics of adding new values to an AVL tree. You should study the instructional resources on AVL trees implementation before attempting this activity.

You will need the following source code files to complete this activity.

- `AvlTree.java`
- `AvlTreeClient.java`

Note: This activity utilizes jGRASP Viewers, which are available in jGRASP, IntelliJ, and Eclipse.

Adding values, self-check

1. Open `AvlTreeClient.java` in jGRASP and compile it.
2. Run this program, observe the output, and make sure you understand what it is doing.
3. Set a breakpoint on the statement `avl.add(value)` in the `main` method.
4. Start the debugger and wait until execution is paused at the breakpoint.
5. Raise a viewer on the `avl` object. Use the viewer settings menu to customize the viewer as you like. For example, you may prefer to have the nodes rendered as circles, and you will most likely prefer that the root be at the top with descendants below.
6. With pen and paper, add the current `value` to an AVL tree that you draw yourself.
7. Step over the call to `add` to confirm your answer.
8. Repeat this process for each element in the `values` array.
9. Make sure you understand how to build an AVL tree from a given sequence of values.

Adding values

1. Open `AvlTreeClient.java` in jGRASP and compile it.
2. Run this program, observe the output, and make sure you understand what it is doing.
3. Set a breakpoint on the statement `avl.add(value)` in the `main` method.
4. Start the debugger and wait until execution is paused at the breakpoint.
5. Open a new Canvas window.
6. Add viewers for `values` , `value` , and `avl` to the canvas window.
7. Step in to the call to `add` .
8. Using step-over and step-in as needed, explore the execution of the `add` and `put` methods, observing their effect in the canvas window.
9. Repeat this process for each element in the `values` array.

10. Make sure you understand how the `add` and `put` methods work, especially the rebalancing operations that happen.

Submission

The submission page for this activity asks you to apply your understanding of AVL trees to a problem and then submit it for a grade.