

Lab 11

Write a `AVLTee` that extends your `LinkedBinaryTree`

1. Add a `parent` reference and a `height` instance variable to the `Node` class
2. Modify/override the rest of your class so that `parent` and `height` are set correctly on insertion and deletion. You might need additional helper methods (to compute height, for example).
3. Implement `rotateLeft` and `rotateRight` and call them appropriately on insertion.
4. Create an `AVLTree<String>` and insert the exercise example given in class, i.e. “M”, “N”, “O”, “L”, “K”, “Q”, “P”, “H”, “I”, “A” and the final balanced tree should look like this:

