

# Spring 2016, CMPE 160, Project 3

Cagatay Yildiz, Yigit Yildirim, Tuna Tugcu

Deadline: May 31, 11.59 pm

In this project, you are going to implement an AVL Tree. Below are the files provided:

**AVLTreeInterface.java:** This interface contains signatures of all methods that you are expected to implement. See the javadoc for more detail.

**AVLTree.java:** This is the main file of the project, where you represent an AVLTree. It implements AVLTreeInterface.java and contains an inner class, which is already implemented. Initially, this class produces errors just because it does not implement methods that it promises to.

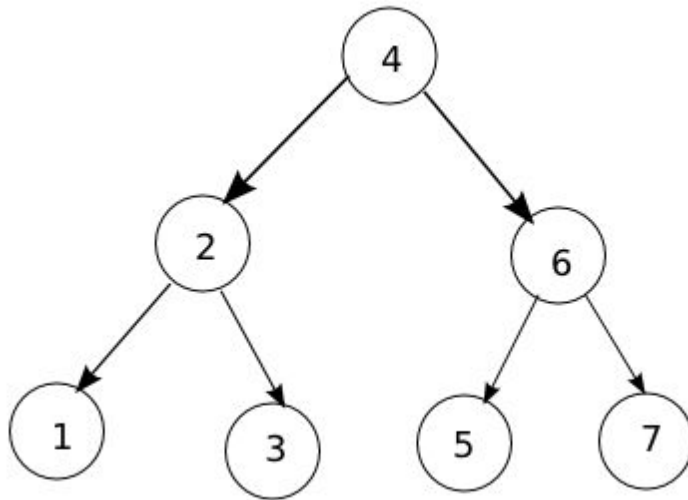
## Details of Submission:

- The way your project is going to be graded is as follows: We will take **AVLTree.java** that you implemented, put it into a separate project and run test cases on this project. So, all the code implemented in different .java files will be basically lost. Recall this while implementing. Note that this is not to say you are allowed to break OOP hierarchy, use meaningless methods, fields, inner classes, etc.
- An example tree and outputs of some of the methods are given in the next page. Please see the code first and try to grasp outputs.
- Do not use any Turkish character in variable/field/method/class/package naming. This is a very bad practice. One of the most common naming conventions for Java programming states
  - Names of variables start with lower-case letters, and use upper-case letters at the start of each word in the name ("camel-style"): e.g., dartsInCircle.
  - Names of classes start with upper-case letters, and use other upper-case letters at the start of each word in the name: e.g., StringTokenizer.
  - Names of public static final fields are written in all upper-case, and use underscores to separate each word in the name: e.g., SPEED\_OF\_LIGHT.
  - See [here](#) for more details regarding proper style of coding.
  - Make sure that your insert and delete methods work properly. I will use those while building your tree. So, for example, if they do not work well but isEmpty() method is okay, you are very likely not to receive any points from test cases for isEmpty() .
- Address your questions directly to the mail list. Also, read the javadoc and mail list carefully. We will not reply a question that has already been answered!

**Good Luck!**

Image from: <http://anoopsmohan.blogspot.com.tr/2011/11/avl-tree-implementation-in-c.html>.

All of the methods below are executed on this tree:



- `tree.bfTraverse()` ---> an ArrayList containing [4,2,6,1,3,5,7]
- `tree.areCousins(2,6)` ---> false
- `tree.areCousins(1,5)` ---> true
- `tree.numElementsInRange(2,8)` ---> 5
- `tree.balanceFactor(4)` ---> 0