# Homework 3

In this homework you will implement a AVL tree (Adelson-Velsky-Landis). The goal is to maintain the AVL property, and to count the number of times a key is inserted into the tree.

The base code for your project is given on with this file. You are required to fill in the functions given in AVL.java. Specifically, you are to fill in the following methods with obvious meanings defined in AVL class:

(1) Insert
(2) Delete
(3) Find
(4) Min
(5) Max
(6) Depth
(7) Print

To implement these methods, you may need private auxiliary methods. Do implement them if you think such helper methods are necessary. Make sure your application does not crash on several circumstances (e.g. if the key to be added already exists)

- University of San Francisco's Data Structure Visualizations tool may help you both while studying and building your homework
  https://www.cs.usfca.edu/~galles/visualization/Algorithms.html

- You have to write a report with the name "Report_HW3.pdf" explaining your HW and which environment you run (Eclipse, NetBeans, for example). The person who reads your report can easily use the class you have written.
- Submissions should be in the form of a zip/rar. When extracted, the result should be a single folder with the name "Group_ID" (where ID is the number of your group which assigned for your cluster.
- Don't forget to put your project and report into the zip/rar file. Report is the 10% of
- project grades.
- All assignments will be checked via Moss (for more info https://theory.stanford.edu/~aiken/moss ). Therefore be honest and do your assignment with only your group.
- Add a 3x2 table in your report and explain contributions as a percent of each member of a group. Example :

| Dilara Kaan | Hercan | Halit Kaan |
|-------------|--------|------------|
| 30%         | 30%    | 40%        |

- Submit your homework on both Mergen and ceng.eskisehir.edu.tr
- Feel free to ask further questions to the teaching assistant (ahmetmuratturk@eskisehir.edu.tr )