# AVL - Exercise

For example, for 10 20 30 40 50 25

```
INSERT 10
10
INSERT 20
10 20
INSERT 30
CASE 3. RIGHT RIGHT
Left rotation in node 10
20 10 30
INSERT 40
20 10 30 40
INSERT 50
CASE 3. RIGHT RIGHT
Left rotation in node 30
20 10 40 30 50
INSERT 25
CASE 4. RIGHT LEFT
Right rotation in node 40
Left rotation in node 20
30 20 40 10 25 50
```

**Task 1.** **Implement the methods:**

Node* **rightRotate**(Node* y) and
Node* **leftRotate**(Node* x)

***(Use Slide 12 as a help)***

**Task 2.** **For each insertion of node with key k, print the following in the console:**

**INSERT k**
**CASE <number and name of the case>.** *(if a case and rotation was needed)*
**LEFT ROTATION IN NODE <key of the node>.** *(if a left rotation is done)*
**RIGHT ROTATION IN NODE <key of the node>.** *(if a right rotation is done)*
***Call LevelOrder to print the tree after the insertion of k***

***Task 3.*** ***Execute the program with:***

```
std::vector<int> data = {90, 80, 70, 60, 50, 75};
```

**Draw in a paper how the tree looks like after each insertion of data[i].**

**Help yourself with the console output.**