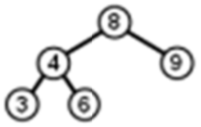
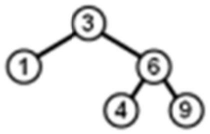
1. Which rotation must be performed after inserting the given key into the given AVL tree?

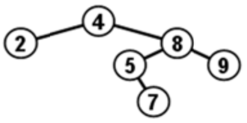
key: 5

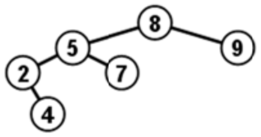


key: 7



2. A key was inserted into the given AVL tree. Which rotation has to be performed now?





3. The keys 40 20 10 30 were inserted in this order into an originally empty AVL tree. The process resulted in a) one R rotation, b) one L rotation, c) one RL rotation, d) one LR rotation, e) no rotation at all.

4. True/False: Simple right rotation in node *u* decreases the depth of the right child of *u*.

5. Draw an AVL tree T containing 8 integer keys. The additional condition is that the operation Insert(19) in T performs also a) LR rotation in the root of T, b) LR rotation in a node of T which is not the root.

6. Consider the shapes of all AVL trees with 8 nodes. what is the number of all possible shapes?

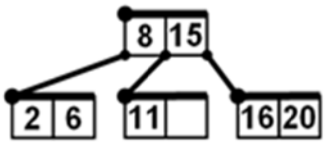
7. The code is intended to perform left rotation in an AVL tree but it is not completely correct. Suggest the correction.

```
Node leftRotation( Node node ) {
    if( node == null ) return node;
    Node p1 = node.right;
    if( p1 == null) return node;
    node.right = p1.left;
    p1.left = node;
    return p1;
}
```

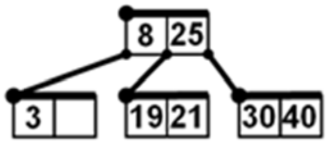
8. Professor Faulterr was heard to claim that any time an AVL tree is balanced by some rotation (single or double one) the depth of the whole tree is decreased. Find a counterexample to this claim.

9. We remove key X from AVL tree using the Delete operation and immediately after that we insert X into the tree again using the Insert operation. We compare the original and the final shape of the tree and conclude: 1. The two shapes will be always different. 2. The two shapes will be always the same. 3. The two shapes may or may not differ depending on the choice of X. Which conclusion is correct?

10. We insert keys 14 and 10 into the given B-tree. What keys will the root contain after this operation?



11. We insert keys 7 and 5 into the given B-tree. What keys will the root contain after this operation?



12. Build a B-tree of order 1 by inserting into it the keys 25, 13, 37, 32, 40, 20, 22 in this order. Next, delete the keys 13, 25, 40, 22, 20, 37, 32 in this order. Draw the tree after each insertion and deletion.

13. The order of B-tree is 10 and it contains 100 000 keys. What is the minimum and maximum possible number of nodes of this tree? What is the minimum and maximum possible depth of this tree?