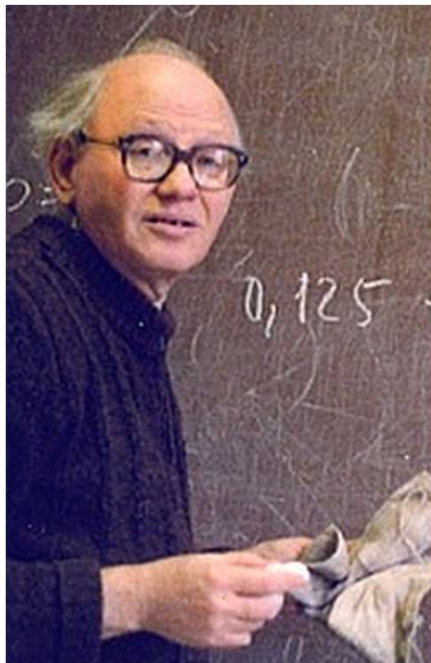


# AVL Tree

# AVL Tree (Bonus material: etymology)

Invented by Georgy **A**delson-**V**elsky and Evgenii **L**andis in 1962



# The AVL Tree Data Structure

An **AVL tree** is a *self-balancing* binary search tree.

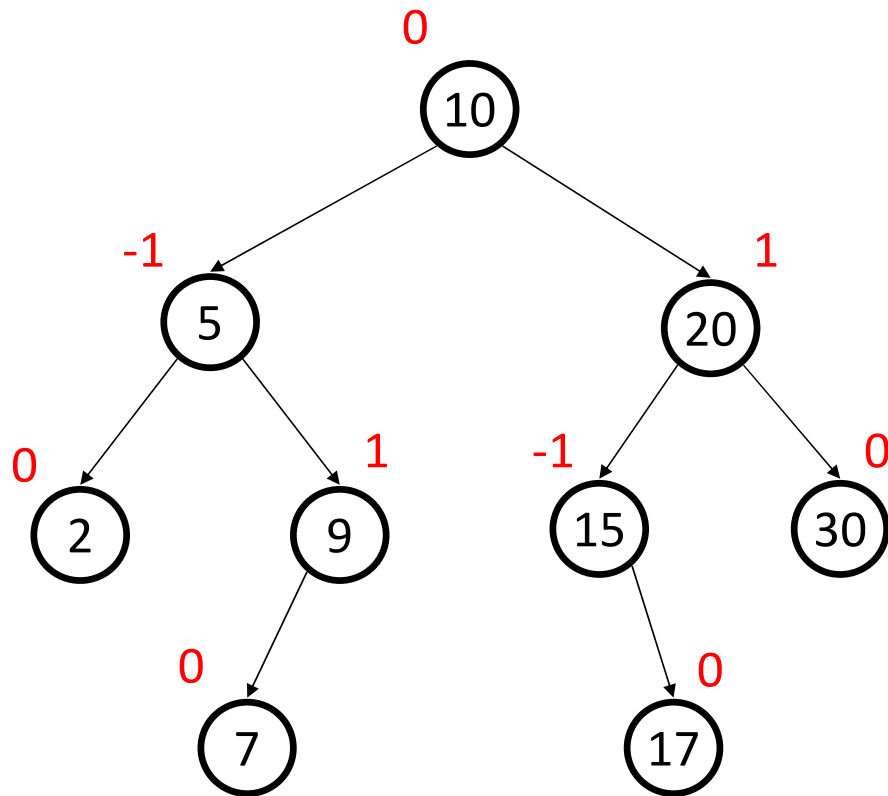
## *Structural properties*

1. *Binary tree* property (same as BST)
2. *Order* property (same as for BST)
3. *Balance condition*:  
balance of every node is between -1 and 1

where  $\text{balance}(\text{node}) = \text{height}(\text{node}.\text{left}) - \text{height}(\text{node}.\text{right})$

Balance factor of a node in an AVL tree is the difference between the height of the left subtree and that of the right subtree of that node. The self balancing property of an avl tree is maintained by the balance factor. The value of balance factor should always be -1, 0 or +1.

# AVL Trees

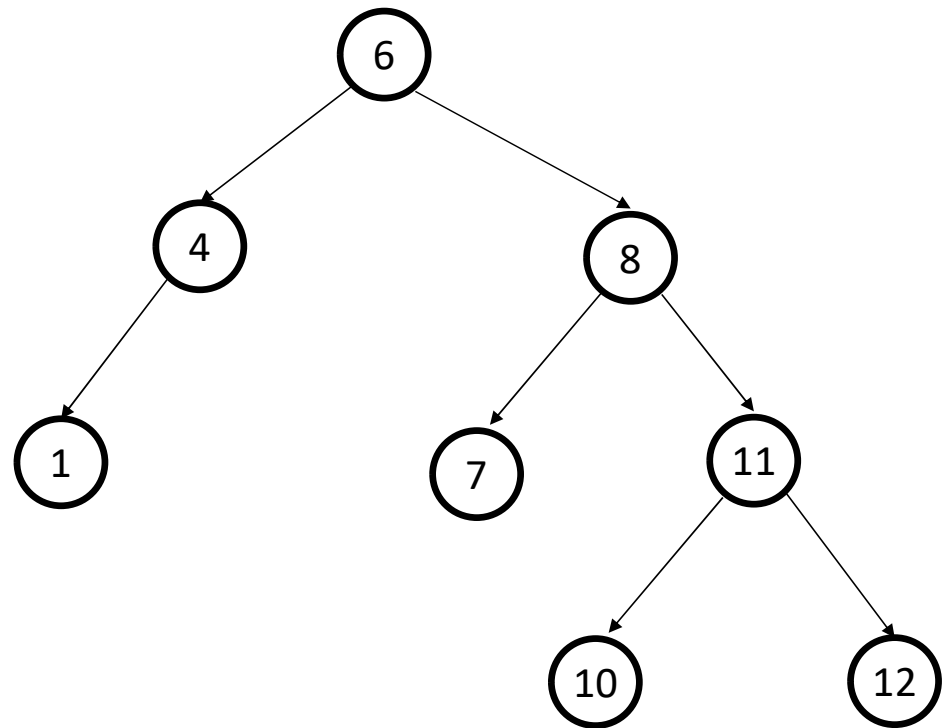


# Example #1: Is this an AVL Tree?

## Balance Condition:

balance of every node is between -1 and 1

where  $\text{balance}(\text{node}) = \text{height}(\text{node.left}) - \text{height}(\text{node.right})$



## Example #2: Is this an AVL Tree?

### Balance Condition:

balance of every node is between -1 and 1

where  $\text{balance}(\text{node}) = \text{height}(\text{node.left}) - \text{height}(\text{node.right})$

