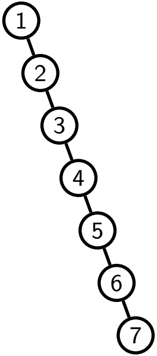
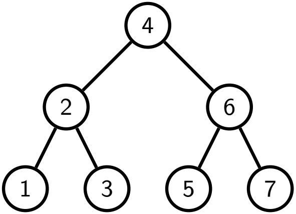
AVL Trees

Balancedness of trees matters



vs.



Can we assume extra conditions to make sure that the height of the tree is under control?

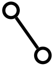
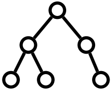
AVL Tree

The **height** of a node is the length of the longest path from that node to a leaf node (compare to the height of a tree)

The **balance** at a node is

$$\begin{pmatrix}\text{The height of}\\ \text{the left subtree}\end{pmatrix} - \begin{pmatrix}\text{The height of}\\ \text{the right subtree}\end{pmatrix}$$

Examples:

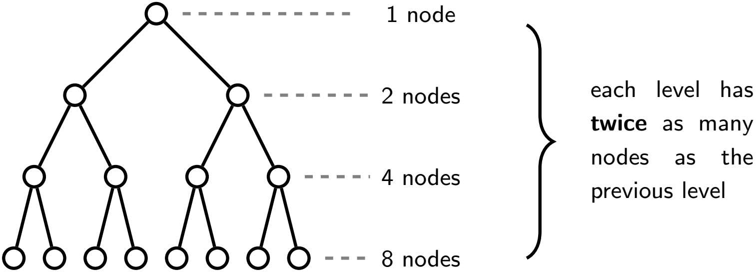
- Note that the height of an empty tree is -1
- The balance at a leaf node is $(-1) - (-1) = 0$.
- The balance at the root of  is $(-1) - 0 = -1$.
- The balance of the root of  is $1 - 1 = 0$.

AVL Tree

Definition: A Binary Search Tree is said to be **AVL** when the balance at *every* node is either 1, 0 or -1.

Perfect Binary Tree = Maximal AVL tree of a given height

Assume that the tree is **perfectly balanced**, that is, the balance of each node is 0. How many nodes does the tree have?



If the tree has height h, then the number of nodes is

$$1 + 2 + 4 + 8 + \cdots + 2^h = 2^{h+1} - 1$$

Another way of saying that the tree is perfectly balanced is that

1. every node, except for leaf nodes, has exactly two children and
2. all leaf nodes are on the same level.