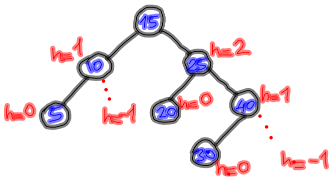
AVL Tree

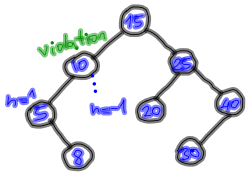
AVL trees are balanced.

An AVL tree is a BST such that for every internal node v of the tree, the heights of the children of v can differ by at most 1.

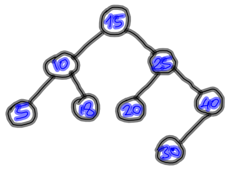
$$|h_l - h_r| \leq 1$$



AVL



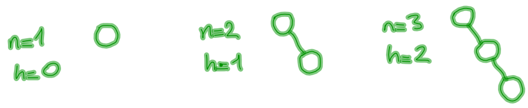
Not AVL

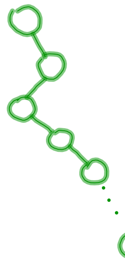


$\rightarrow$ Not AVL, because it is not a BST

Proposition: The height of an AVL tree with n nodes is $\underbrace{O(\lg n)}_{\text{upper bound}}$ $\quad \lg n = \log_2 n$

Question: What is the possible maximum height of a BST with n nodes?





for n nodes

$h_{max} = n-1 = O(n)$

Question: What is the possible minimum height of a BST with n nodes?

1

2^1

2^2

2^3

level h $\quad + \; 2^h$

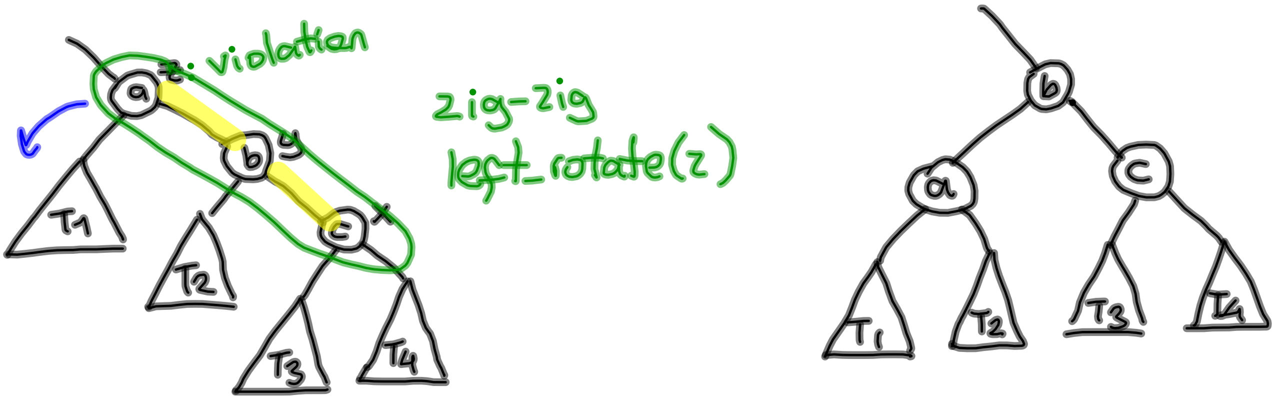
$$1+2^1+\dots+2^h = \frac{2^{h+1}-1}{2-1} \geq n$$

$$2^{h+1} \geq n+1$$
$$h+1 \geq \lg(n+1)$$
$$h \geq \lg(n+1)-1$$

$$h_{min} = \lg(n+1)-1$$
$$h_{min} = \Omega(\lg n)$$

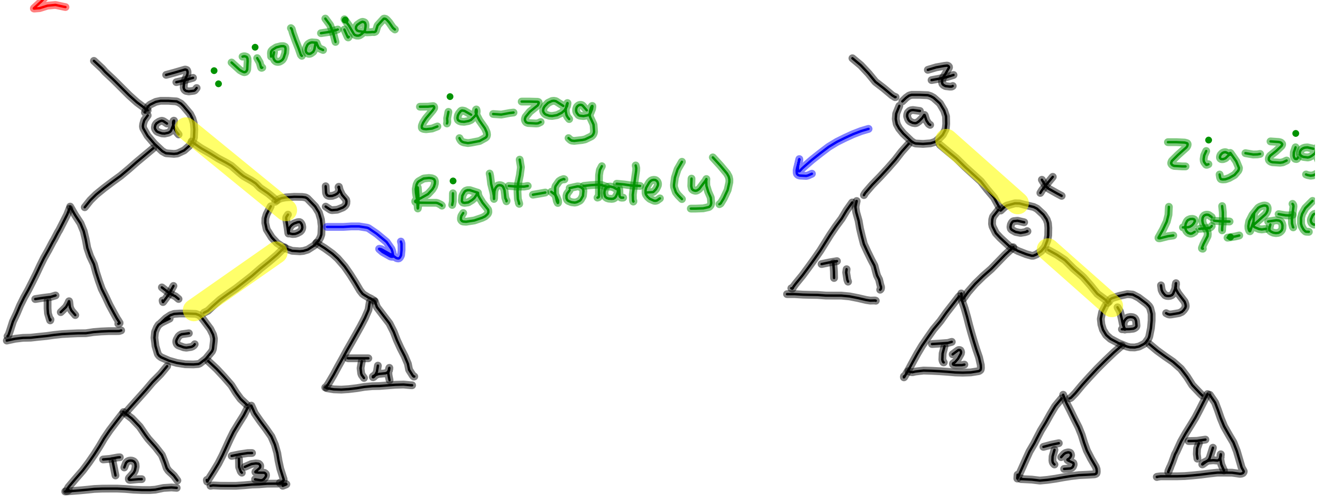
Rotations

1. Single Rotation



$T_1 < a < T_2 < b < T_3 < c < T_4$

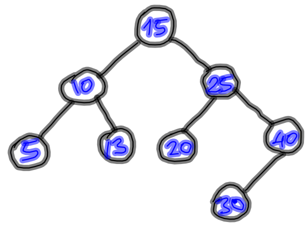
2 Double Rotation



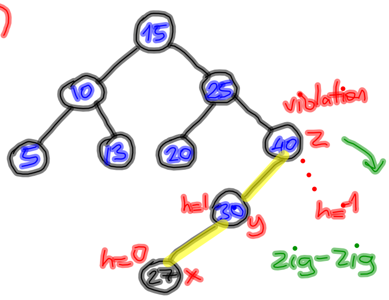
Insertion

Inserting a node into an AVL tree changes the heights of some of nodes in the tree.

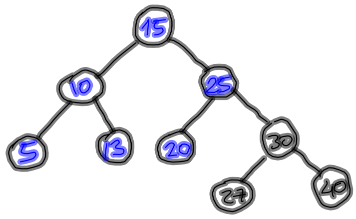
If insertion causes tree to become unbalanced, we travel up the tree from the newly created node until we find the first node x such that its grandparent z is unbalanced node.



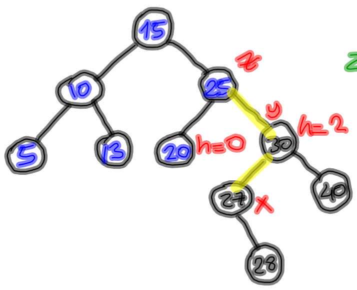
insert(27)



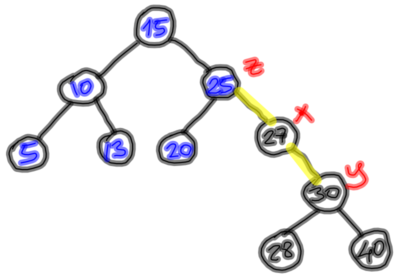
Right_Rotate(z)



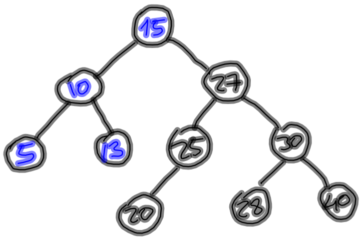
insert(28)



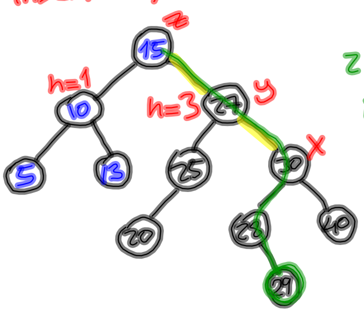
zig-zag
R_R(30)



L_R(25)



insert(29)



zig-zig
L_R(15)