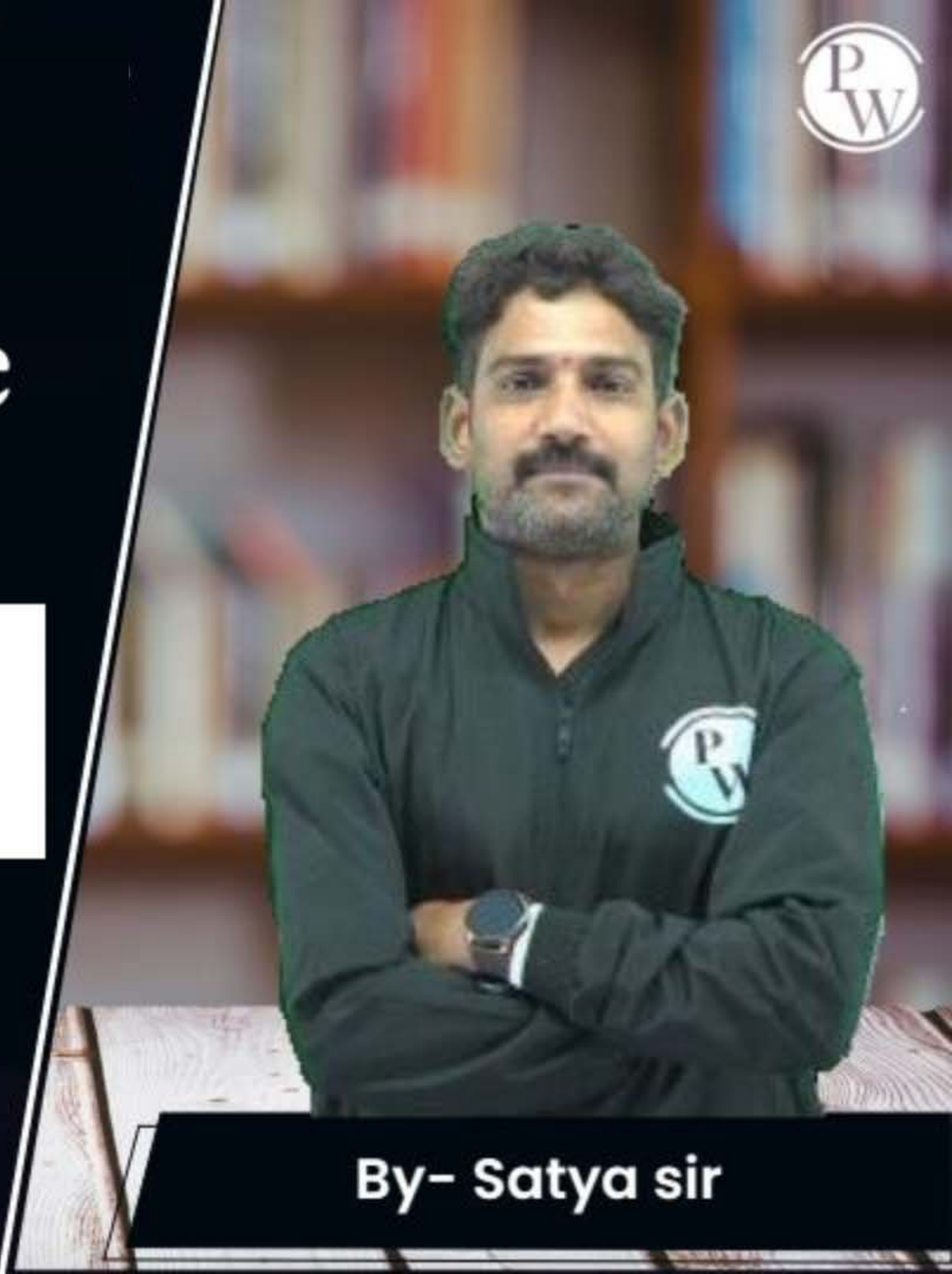
Data Science & Artificial Intelligence

Data Structures Through Python

TREES

Lecture No.- 06



By- Satya sir

Topics to be Covered

AVL Trees

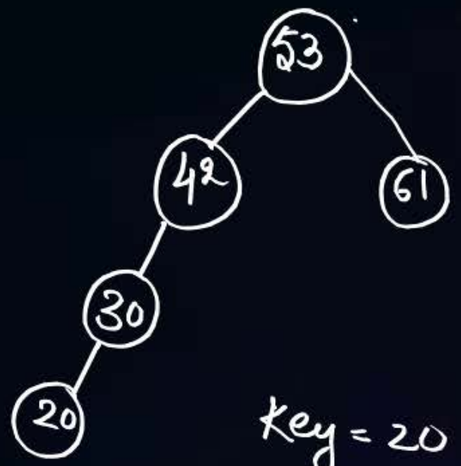
- What is AVL Tree ?
- Balance factor ?
- Height balanced ?
- Rotations

Topic : AVL Trees

Height Numbering from '1'

Let us Construct BST (Left Subtree < Parent < Right Subtree) by Inserting the Nodes in the order :

1) 53, 42, 61, 30, 20

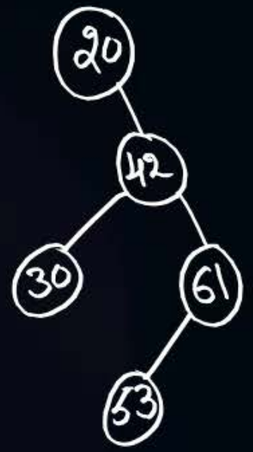


key = 20

height = 4

No. of Comp = 4

2) 20, 42, 61, 53, 30



key = 53

height = 4

No. of Comp = 4

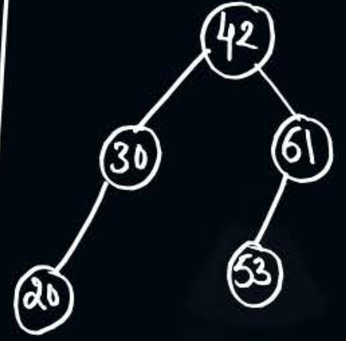
3) 61, 53, 42, 30, 20

key = 20

height = 5

No. of Comp = 5

4) 42, 61, 30, 53, 20

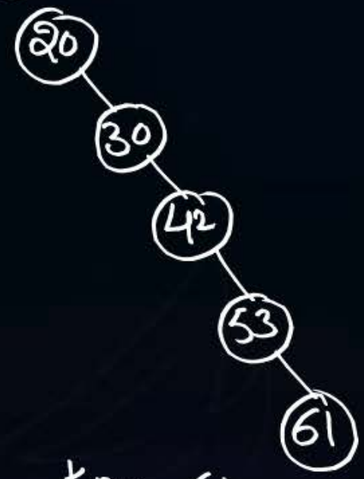


key = 20

height = 3

No. of Comp = 3

5) 20, 30, 42, 53, 61



key = 61

height = 5

No. of Comp = 5

Topic : AVL Trees

- The Time Complexity for Operations on BST Varies as Per the height of BST.
  $[\log_2^n$ to $n]$
- To retain the Time complexity to $O(\log_2^n)$, we need to make/keep height in balanced manner.
- Such "Height balanced BST" is known as AVL Tree.
- A BST is said to be Height balanced BST (AVL Tree) if and only if,
  The balance factors of all nodes is in the range = $\{-1, 0, +1\}$.

Topic : AVL Trees

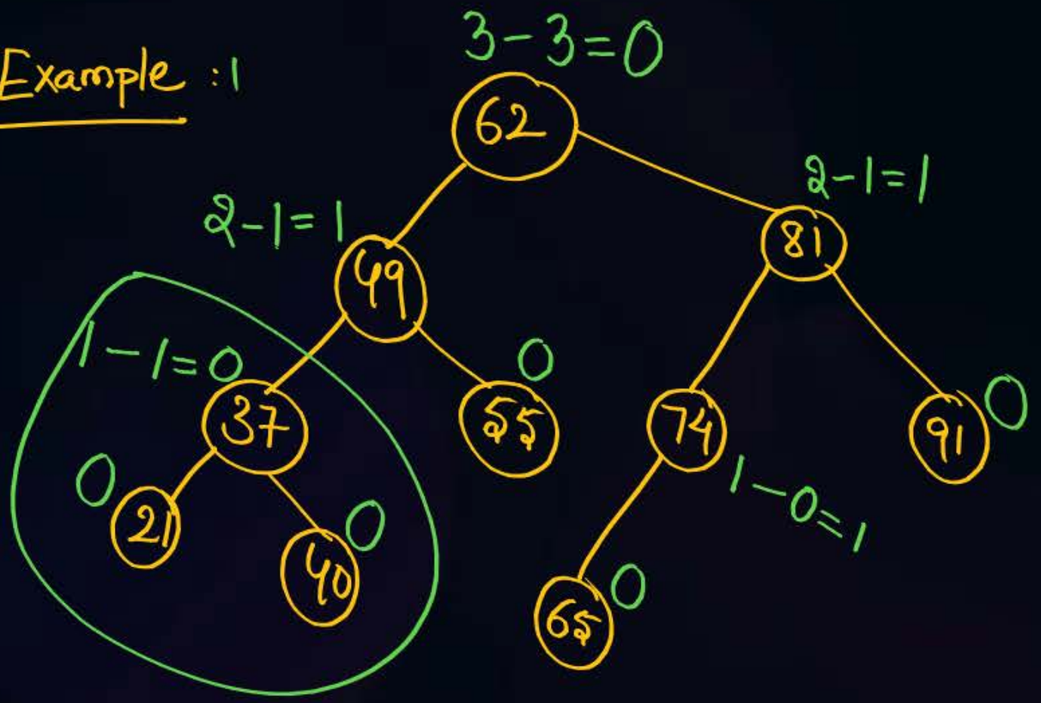
Balance factor

Height Numbering starts from '1'

Balance factor of a Node == [height of Left subtree — height of right Sub tree]

NOTE: Balance factor of any leaf Node = 0.

Example :1



Balance factors = {0, 1, 1, 0, 0, 1, 0, 0, 0, 0}

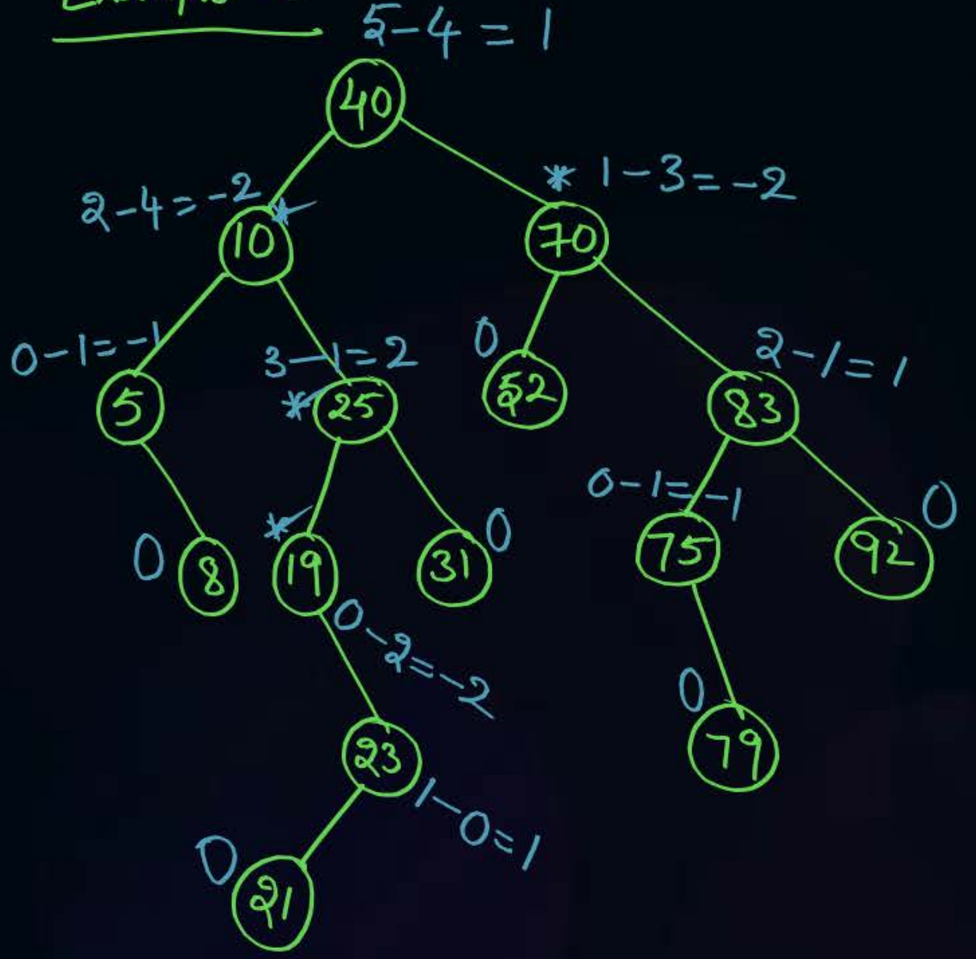
Given BST is Balanced BST == AVL Tree.

Topic : AVL Trees

The Nodes who balance factor is not in the range = {-1, 0, +1} are called as CRITICAL NODES*

If atleast one Critical Node in a BST, it is Not an AVL Tree.

Example-2



Not Height balanced BST.

⇒ Not an AVL Tree.

{1, -2, -2, -1, 2, 0, 1, 0, -2, 0, -1, 0, 1, 0, 0}

Topic : AVL Trees

$n = 65536$ ⇒ Let Each Unit of time = 1msec

⇒ $O(n)$ ⇒ 65536 msec

⇒ $O(\log_2^n)$ ⇒ $O(\log_2^{65536})$

$= \log_2^{2^{16}}$

$= 16$ msec

Need to balance ? To make Time Complexity from $O(n)$ to $O(\log_2^n)$

Topic : AVL Trees

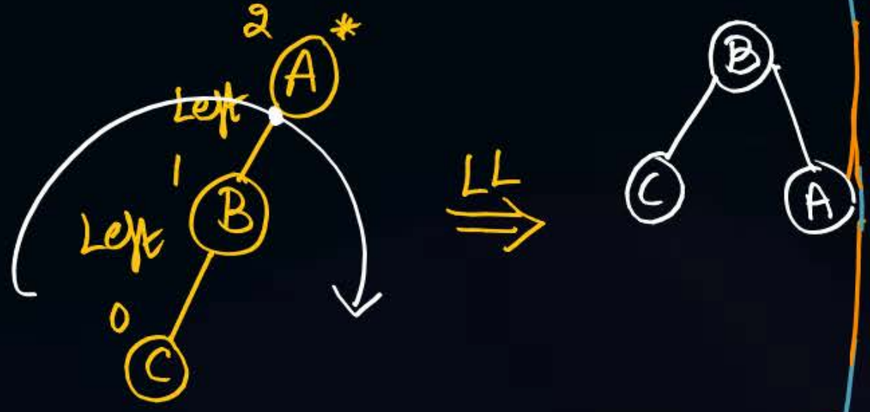
How To Balance?

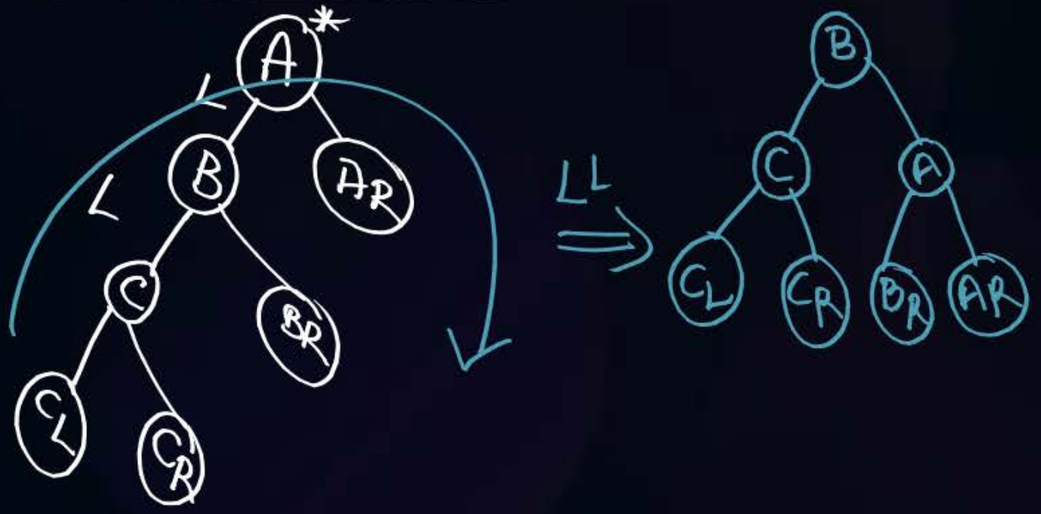
- To Balance a BST, Perform Rotations.
- There are 4 types of Rotations :

1) LL Rotation
2) RR Rotation
3) LR Rotation
4) RL Rotation

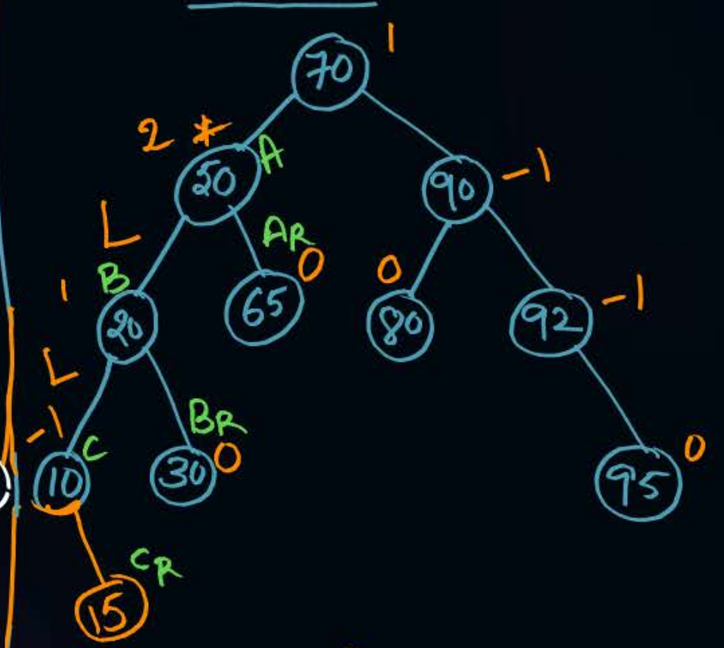
Topic : AVL Trees

LL Rotation



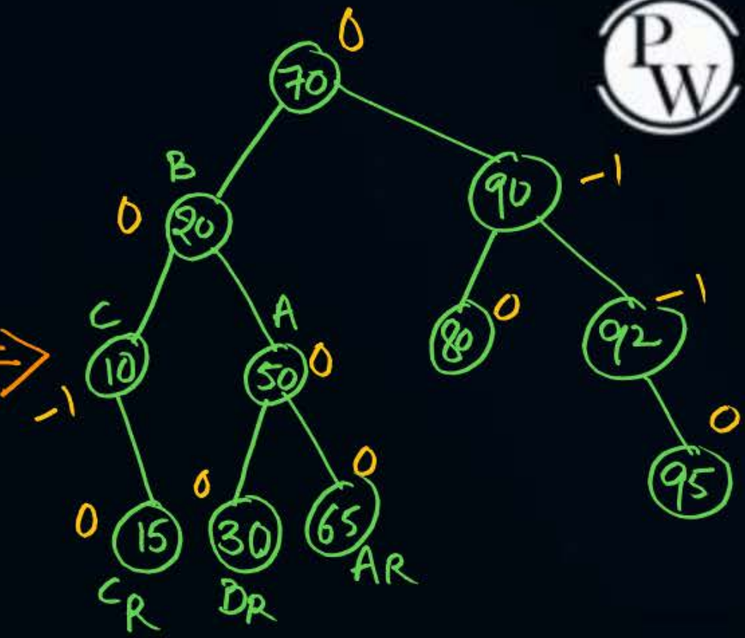


Example



Not Balanced

LL

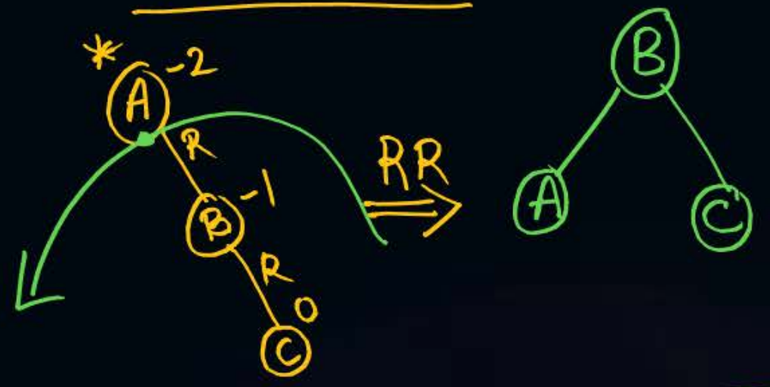


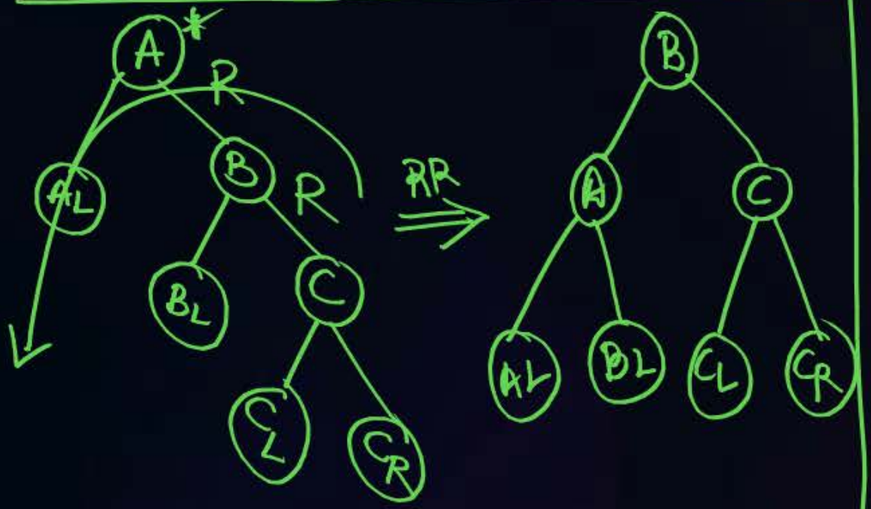
Balanced BST

⇒ AVL Tree.

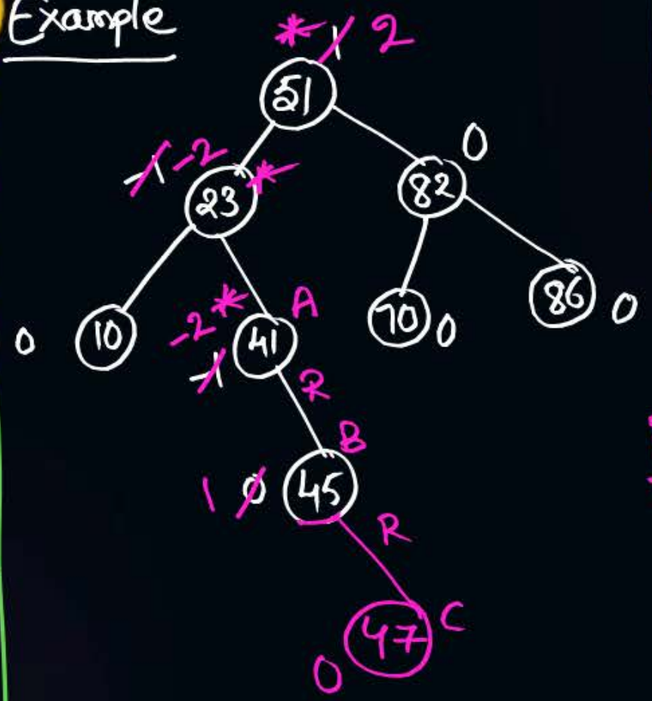
Topic : AVL Trees

RR Rotation





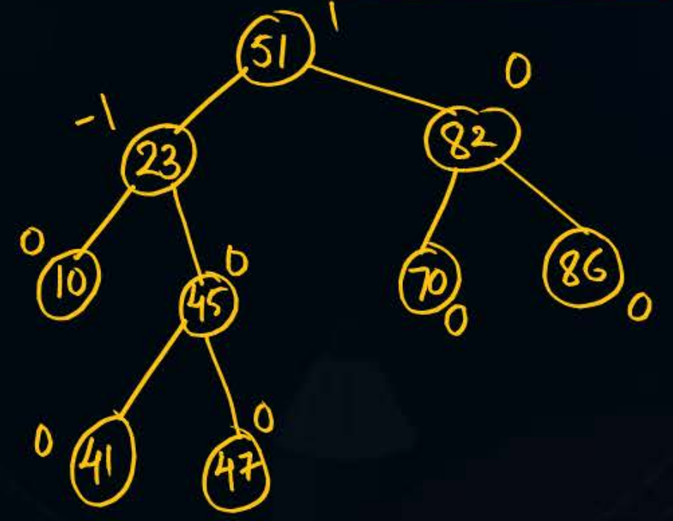
Example



Not Balanced

when Two or more critical Nodes, Bottom most critical Node will have Priority.

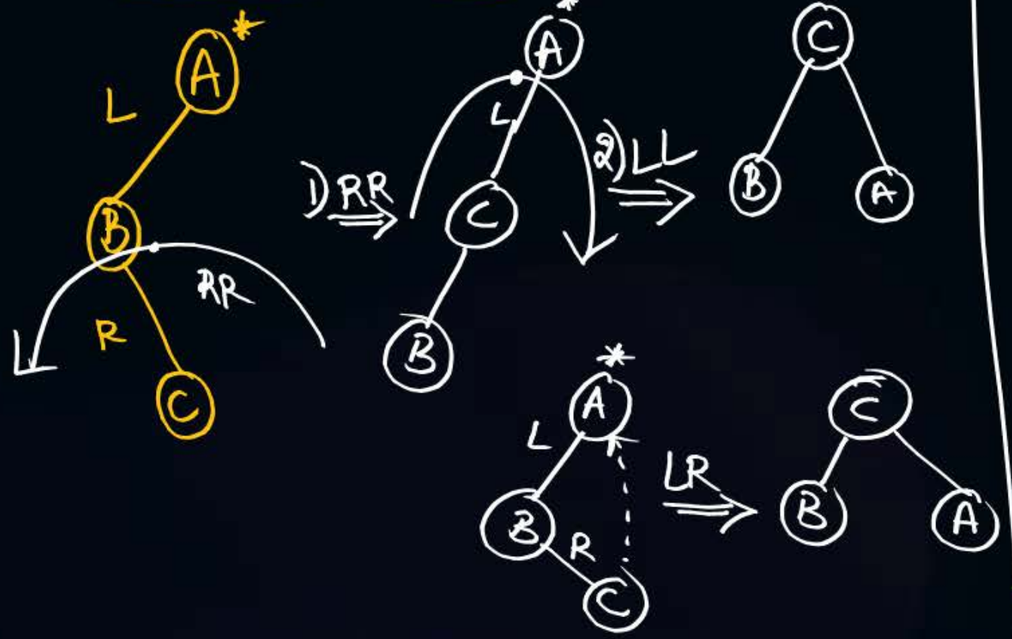
RR

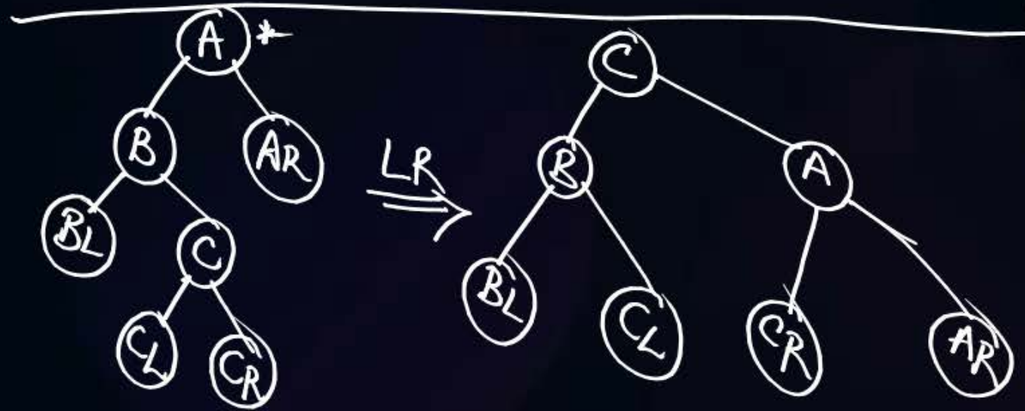


Balanced BST

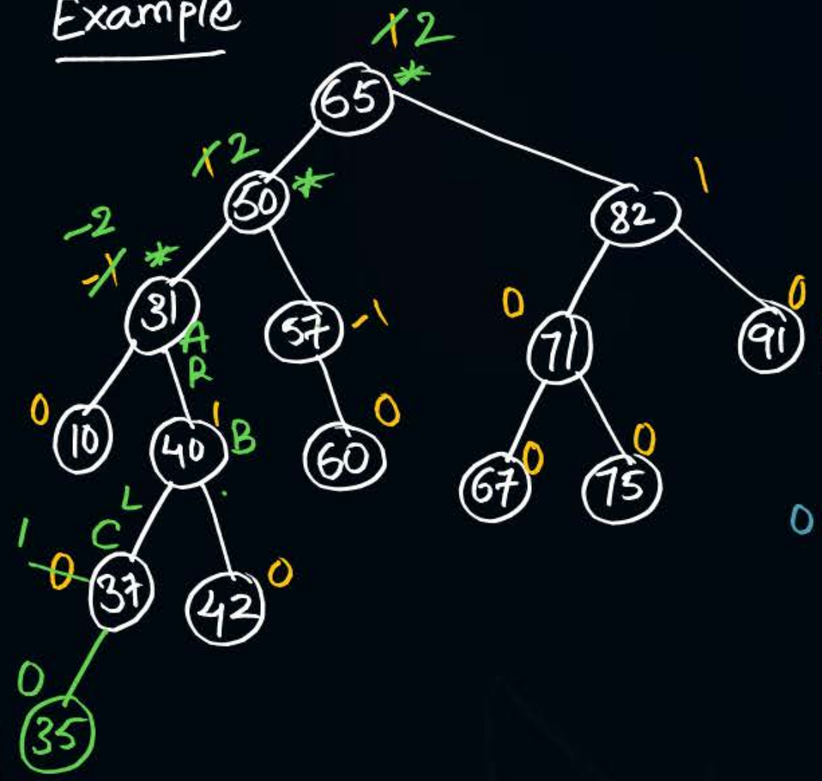
Topic : AVL Trees

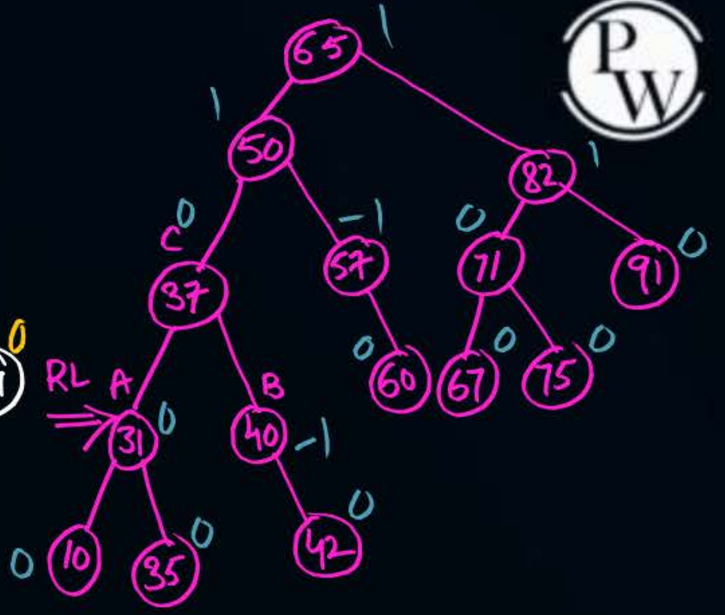
LR Rotation (Double Rotation)



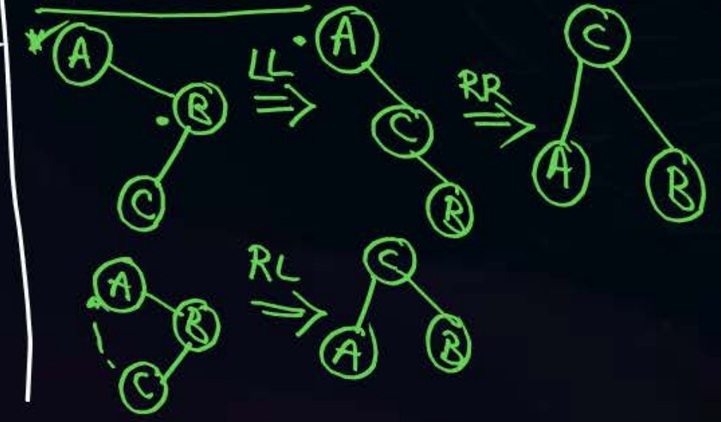


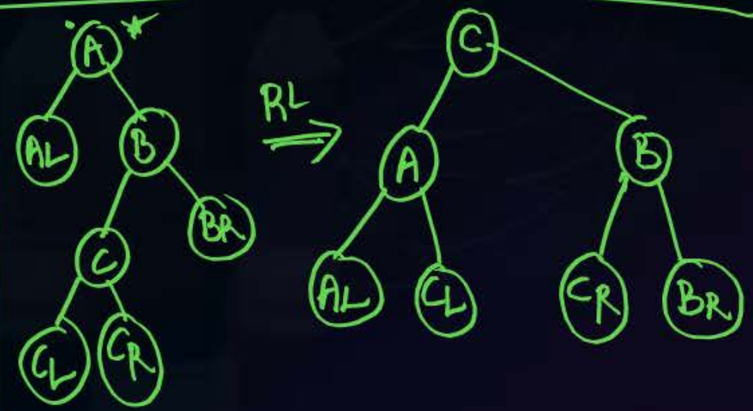
Example





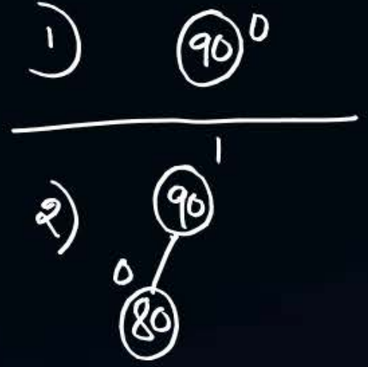
RL Rotation

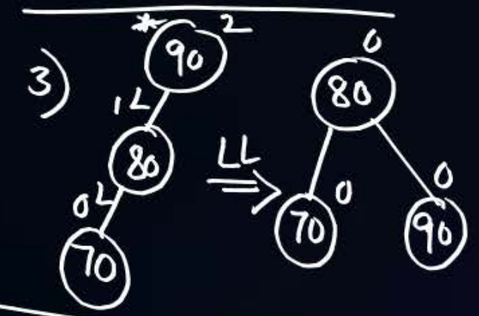


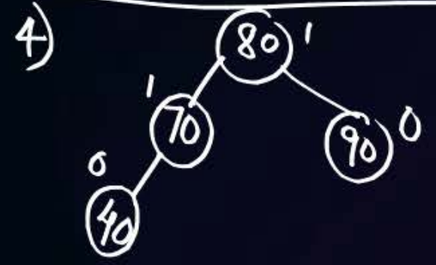


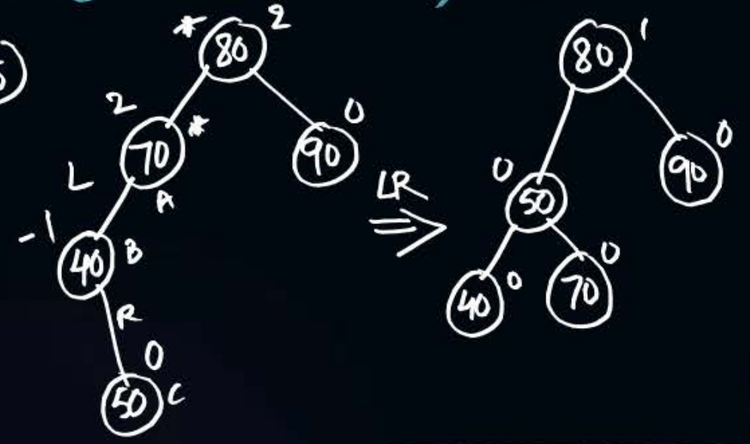
Topic : AVL Trees

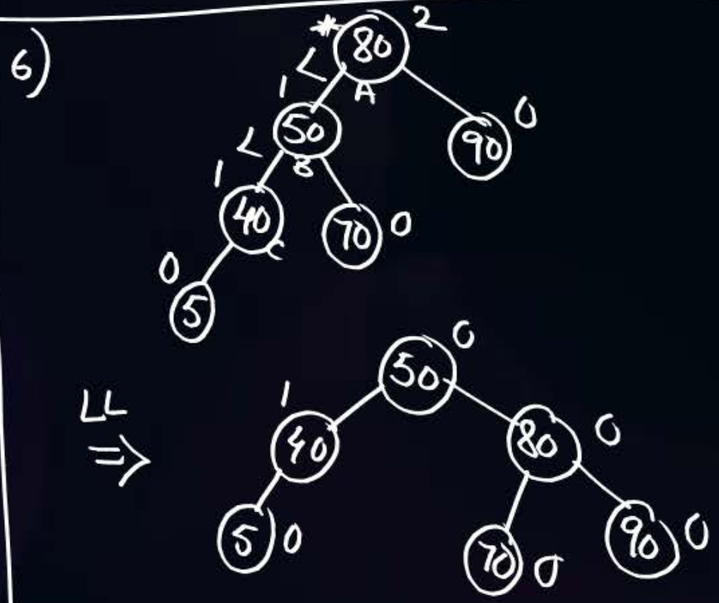
Construct an AVL Tree with Elements, inserted in the order: 90, 80, 70, 40, 50, 5, 32, 47, 45, 49

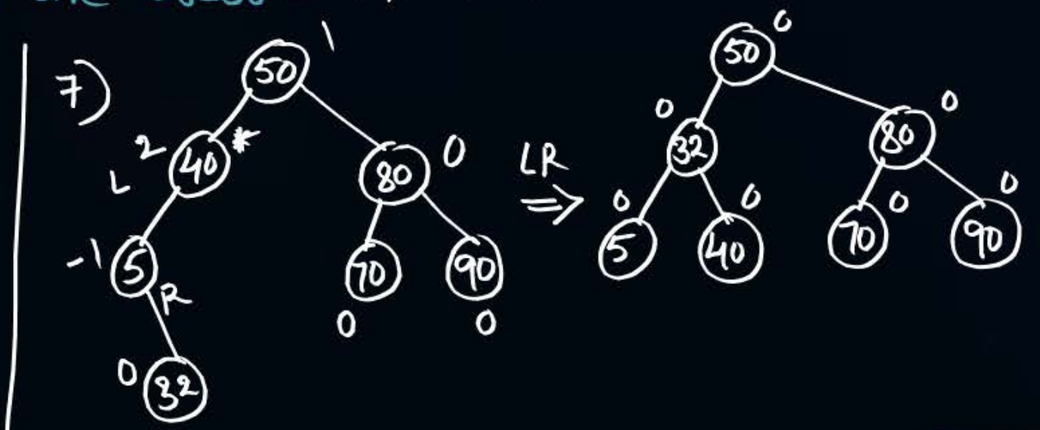


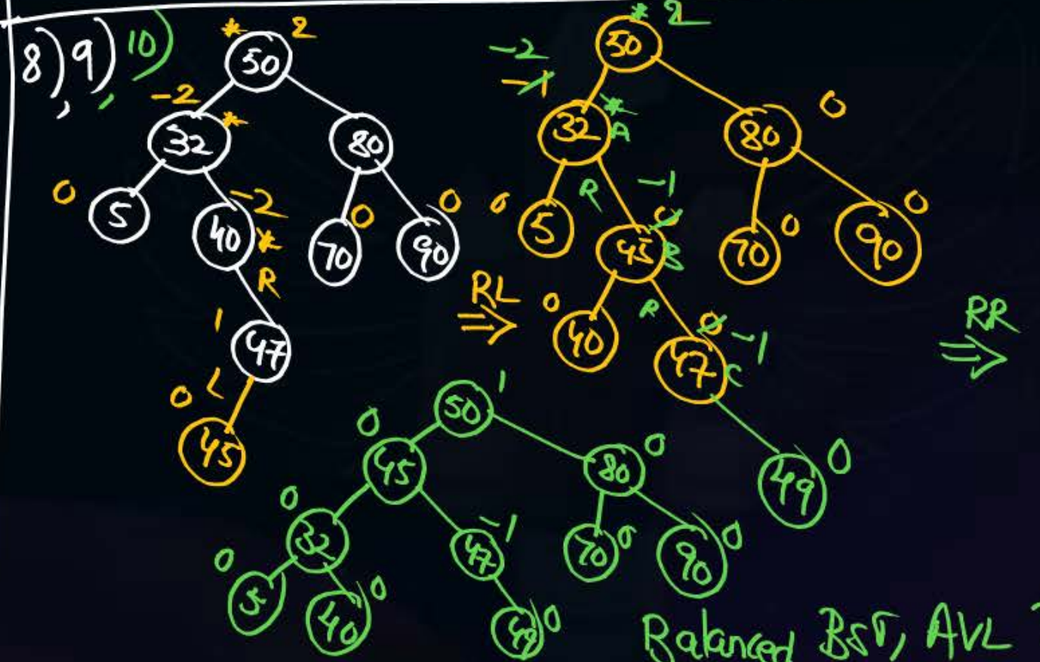












Topic : AVL Trees

- Minimum height of an AVL Tree with 'N' Nodes = $\lfloor \log_2^{(N+1)} \rfloor$

- Maximum height of an AVL Tree with 'N' Nodes = $1.44 * \log_2 N$

Time Complexity of AVL Tree == $O(\log_2 N)$

2 mins Summary

AVL Tree

- Balance factor ?
- Height Balanced ?
- Rotations
  - LL
  - RR
  - LR
  - RL

THANK - YOU