AVL Tree

- balanced factor of every node can be 1, 0, -1
- Leaf's balance factor = 0

* verify given BST is balanced or not:

```
** int isBalanced (BST *bst){
        if (!bst)
            return 1;

        if (abs(leftheight - rightheight) ≥ 1)
            return false;
        return isBalanced(Left) && isBalanced(Right)
```

Ipyoy3

Code this up

Type of imbalance decides rotation

LL → LL

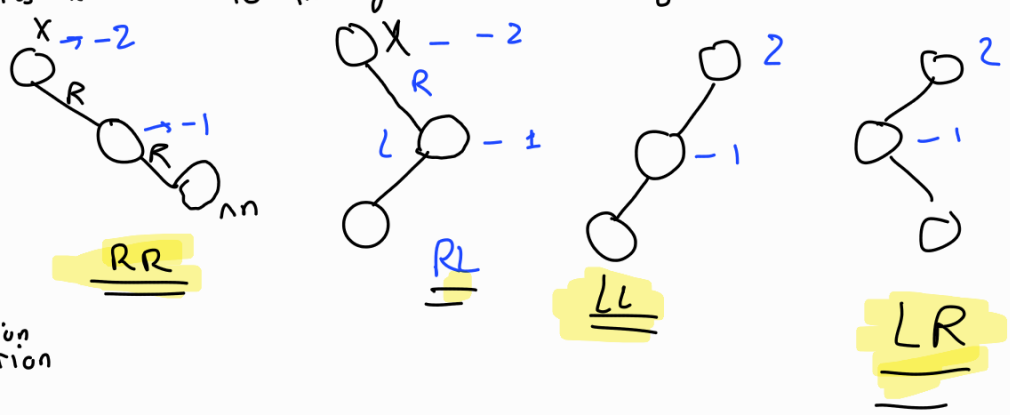
Left Left

RR → RR

Right Subtree

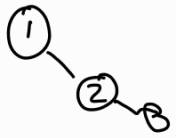
A-node ⟶ Node which is imbalanced

RR: Newly inserted Node is in right subtree of right subtree



LL . 1 Rotation
RR ...) 1 Rotation

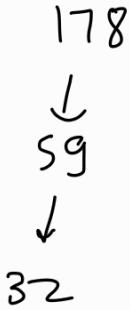
RL → 2 Rotation
LR → 2 Rotations.



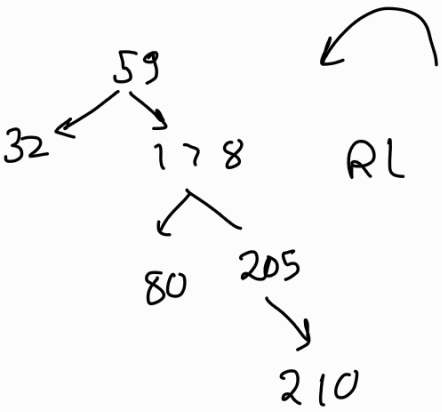
RR Rotation ⟹

178, 32, 59, 205, 210, 80, 75, 66, 100, 125, 10, 15, 108, 64

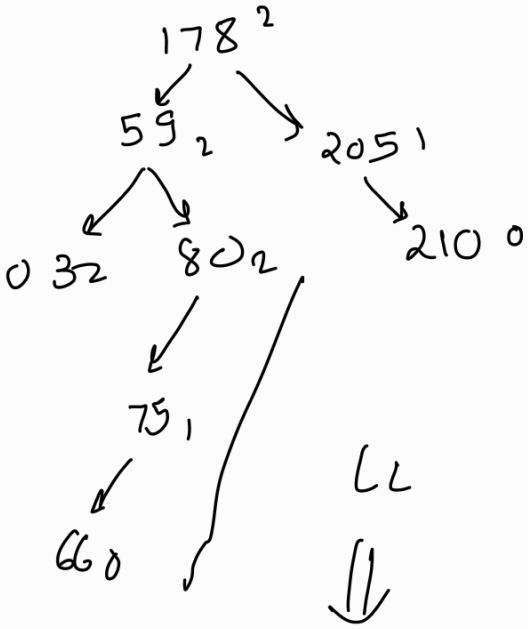
LR ① ⟹



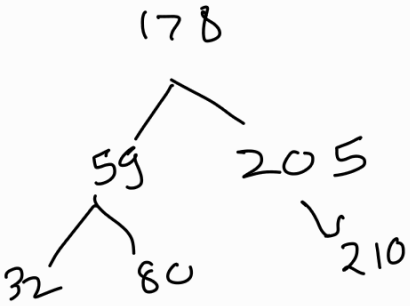
② ⟹



RL ① ⟹

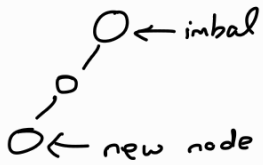


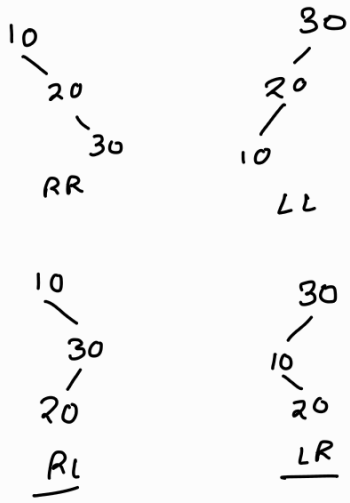
LL ⇓



imbalance Node.

```
if(nn->key < imbal->key)   <---- gives L
    if (nn->key  < imbal->key)
        //LL imbalance
    else
        //LR imbalance
else
    if (nn->key  < imbal->key)
        //RL
    else
        //RR
```

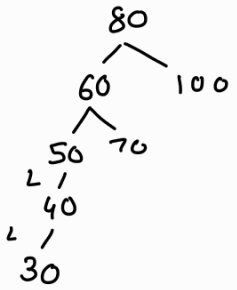


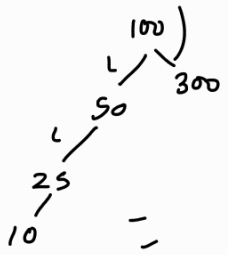


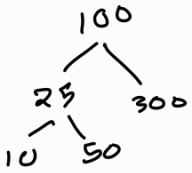

```
void LLrotate (AVLTree *t ,node *A){

    Node*  B = A->left;
    Node * BR = B->right;

    Node * Ap = A->parent ; //Parent of A;
    B->parent = AP ;

    if (AP == NULL){
        *t = B;
    }
    A->parent = B
else{
 if ( AP-> right == A)
     Ap->right = B
 else
     AP ->left = B
}

 A->parent = B

  B->right = A

  A->left = Br

   if (Br){
       Br->parent = A;
   }
```

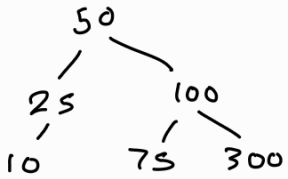
store height in all nodes.

LL causing tree

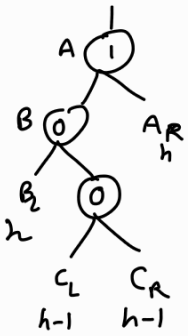


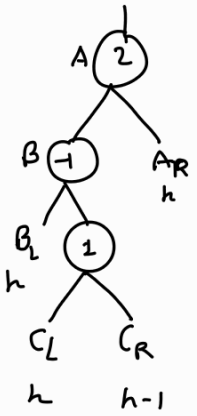


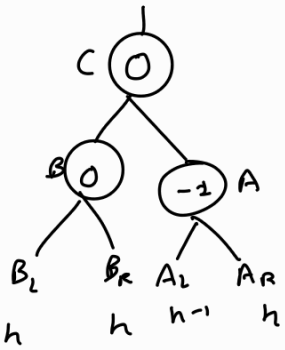


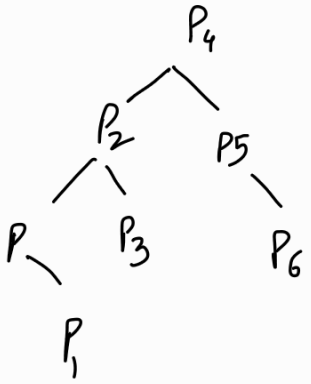


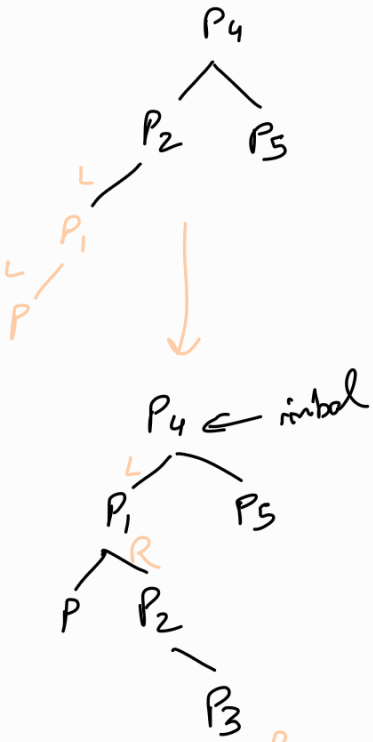
LR Rotation (Case 2):

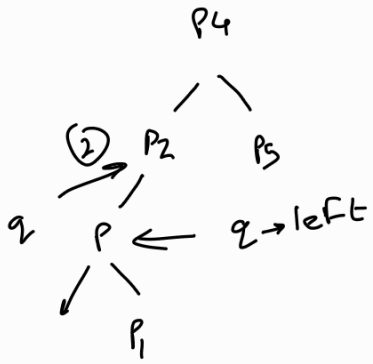


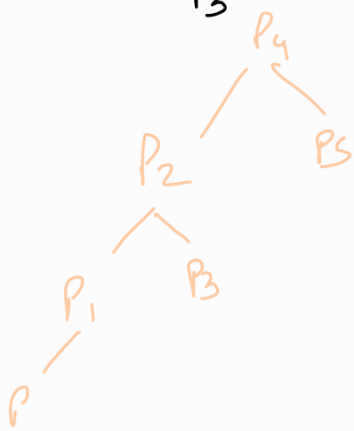




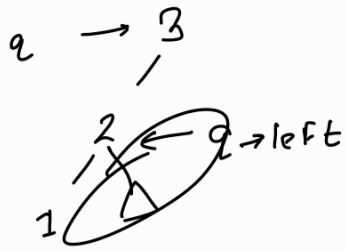

P4
P2
P5
P
P3
P6
P1


P4
P2
P5
L
P1
L
P
P4
initial
L
P1
P5
R
P
P2
P3

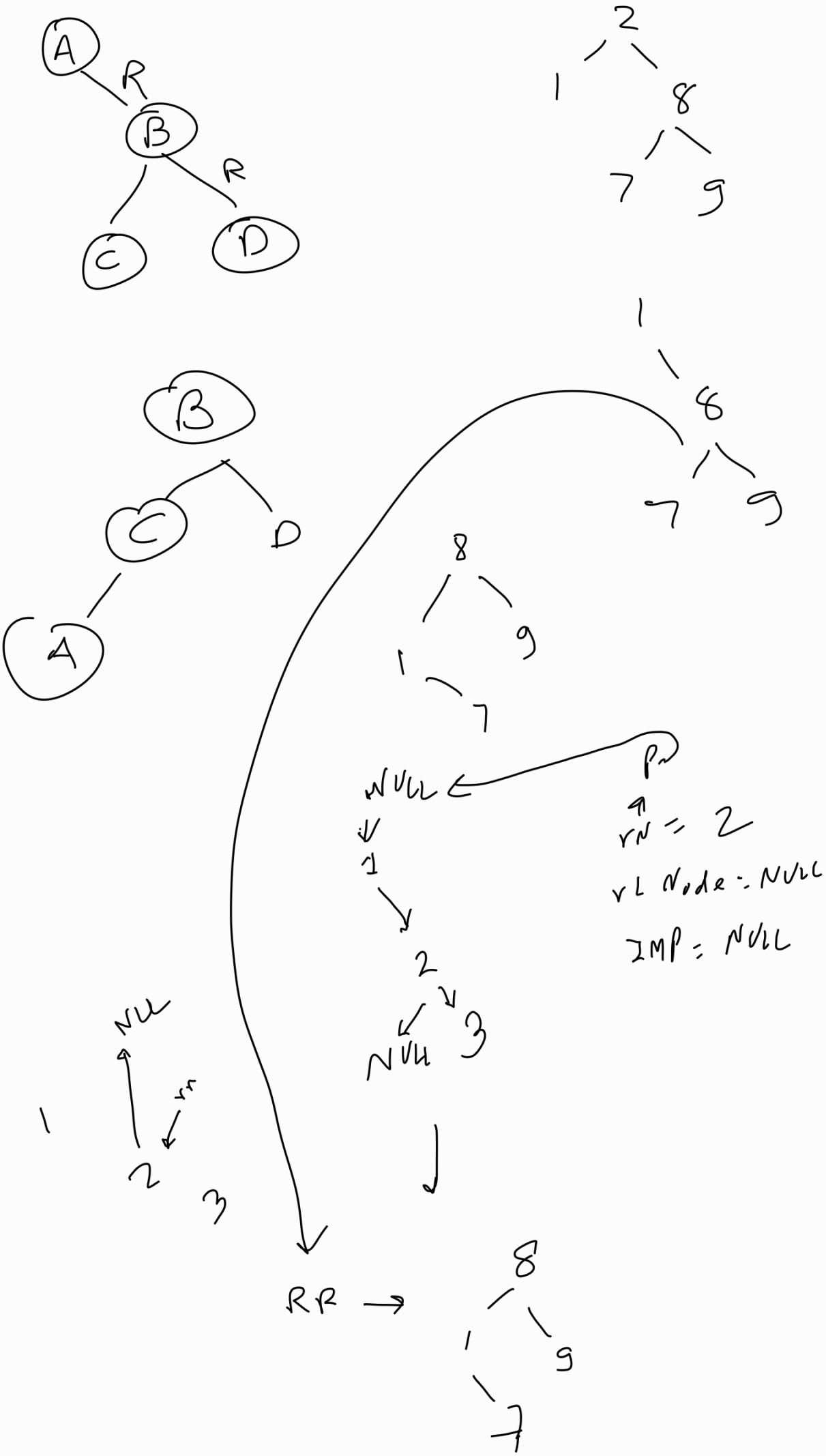

P4
2
P2
P5
q
P
q→left
P1

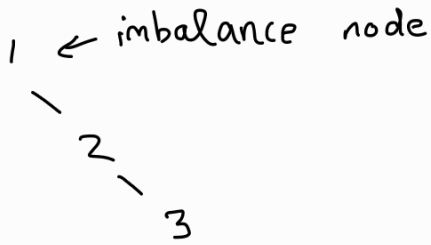

P4
P2
P5
P1
P3
P

LR

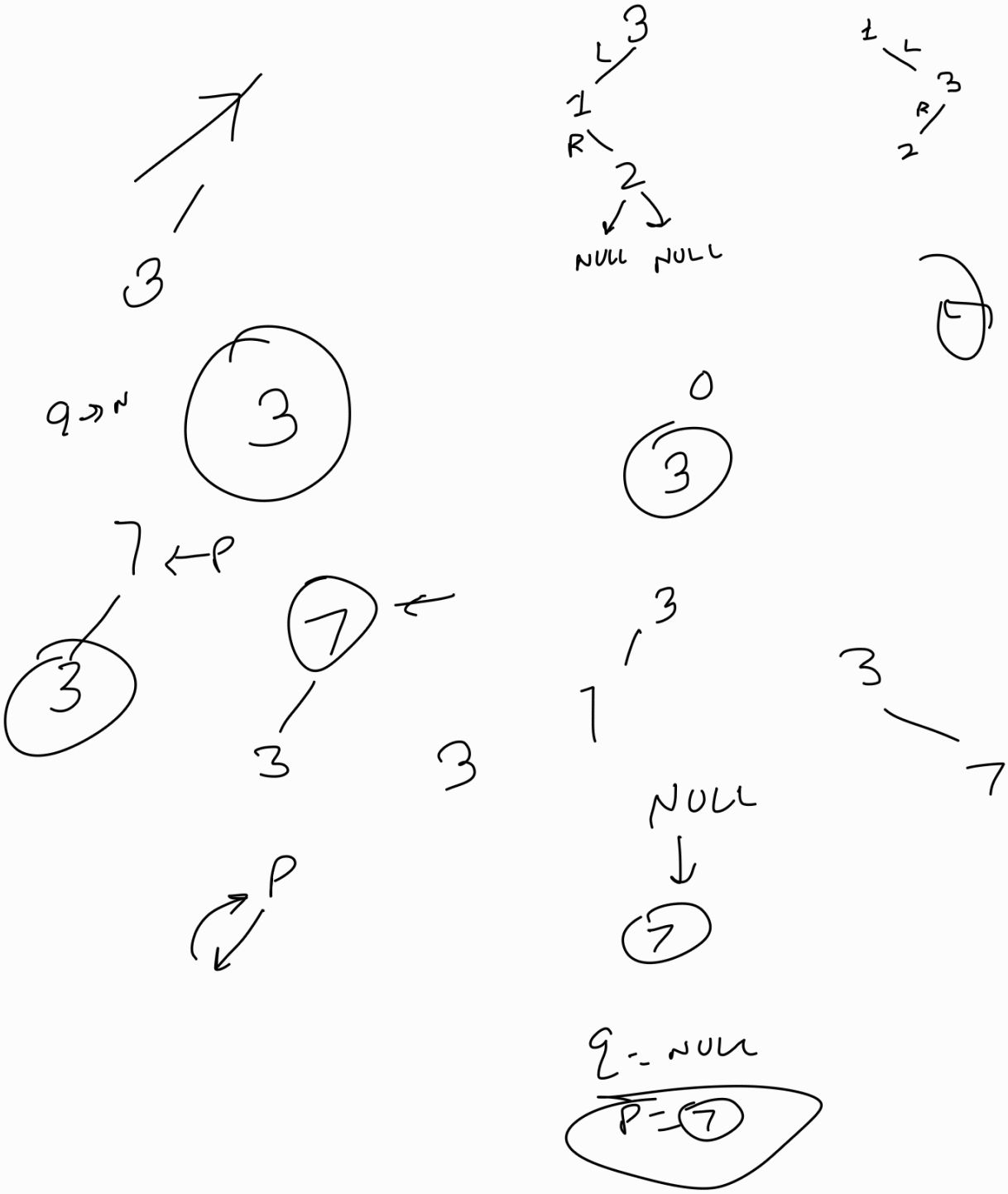

q
3
2
q→left
1

P2
-1
P1
P4
-1
P
P3
P5
-1
P6
0


A
R
B
R
C
D
B
C
D
A
NULL
1
2
3
rr
NULL
1
2
NULL
3
P
rr = 2
rL Node = NULL
IMP = NULL
RR →



imbalance Node = 1
right Node = 2
rightleft Node = NULL;
imbalance Parent = NULL



Max. height of AVL tree $\Rightarrow$ Fibonnachi series

height 0

height 1

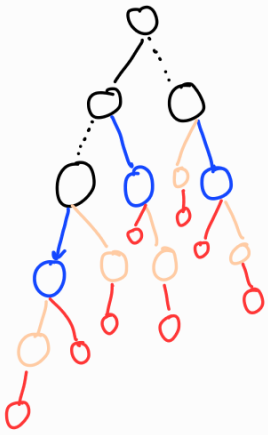
height 2

height 3

height 4

height 5

height 6



Nodes

1

1+1

1+1+2

1+1+2+3

1+1+2+3+5

1+1+2+3+5+8

1+1+2+3+5+8+13

Total number of element to get height 3.

for inserting 20 nodes in a tree,
max possible height == 5

$N_0 = 1$

$N_1 = N_0 + 1 = 2$

$N_2 = N_0 + N_1 + 1 = 1+2+1 = 4$

$N_3 = N_1 + N_2 + 1 = 2+4+1 = 7$

$N_4 = N_2 + N_3 + 1 = 12$

$N_5 = N_3 + N_4 + 1 = 20$

$1.44 \log n$

Let $S(N)$ represent sum of first 'N' terms of fibonacci series. Calculate $(N+2)^{th}$ term and subtract 1 from result.

N^{th} term of series $\Rightarrow$ $F_N = \dfrac{\left(\frac{1+\sqrt{5}}{2}\right)^N}{\sqrt{5}}$

$\phi^i = \left(\dfrac{1+\sqrt{5}}{2}\right)^N$

$\left(\dfrac{1+\sqrt{5}}{2}\right)^5$

0	1	1	2	3	5	8	13	21	34
0	1	2	3	4	5	6	7	8	9

height of AVL tree having 'n' nodes is at most

$$1.44 \log_2(n+2)$$

and is at least $\log_2(n+1)$