Feb 9, 2016

AVL-trees:

Balanced BST: guaranteed $O(\lg n)$ performance for insertion and <u>deletion</u> (search)

AVL-tree: for any node, the height difference between its left and right subtrees is ≤ 1

Recall that for an AVL-tree of n nodes, its height $\leq \sim 1.4 \lg n$

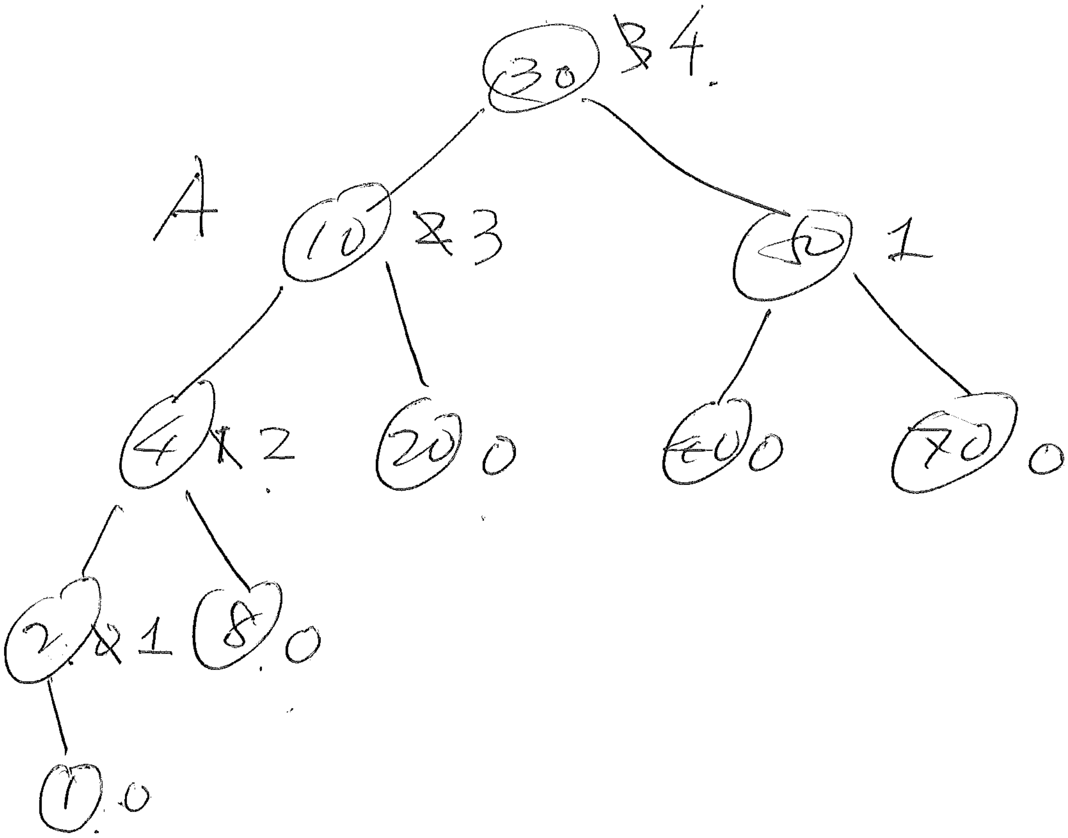
---

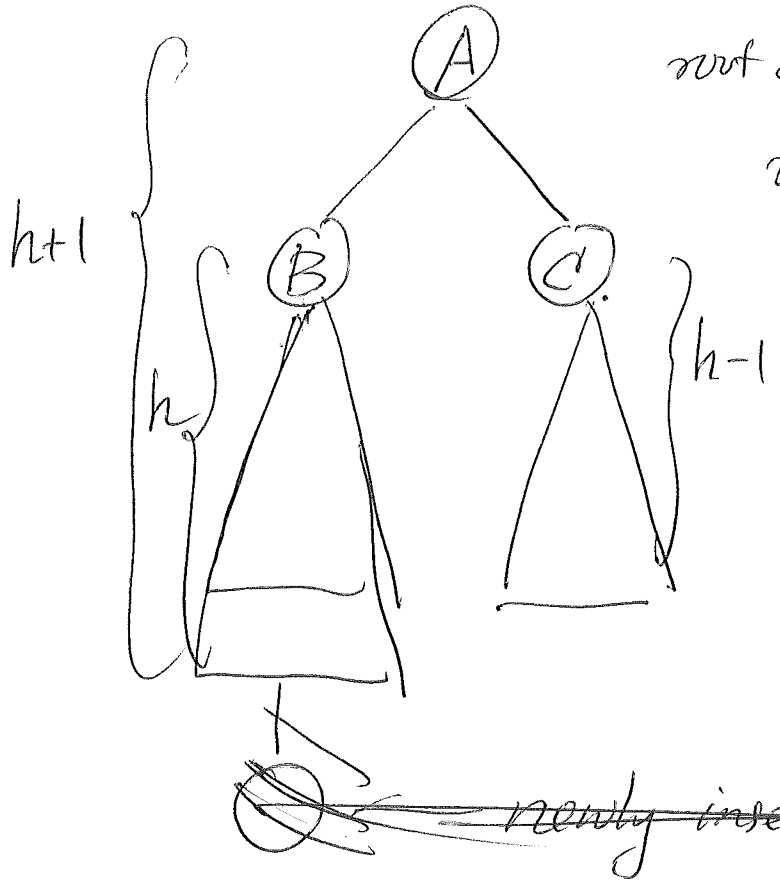
insertion in AVL

① a normal BST insertion

② if the normal insertion cause a violation of the AVL property, we will rebalance the tree with rotation(s)

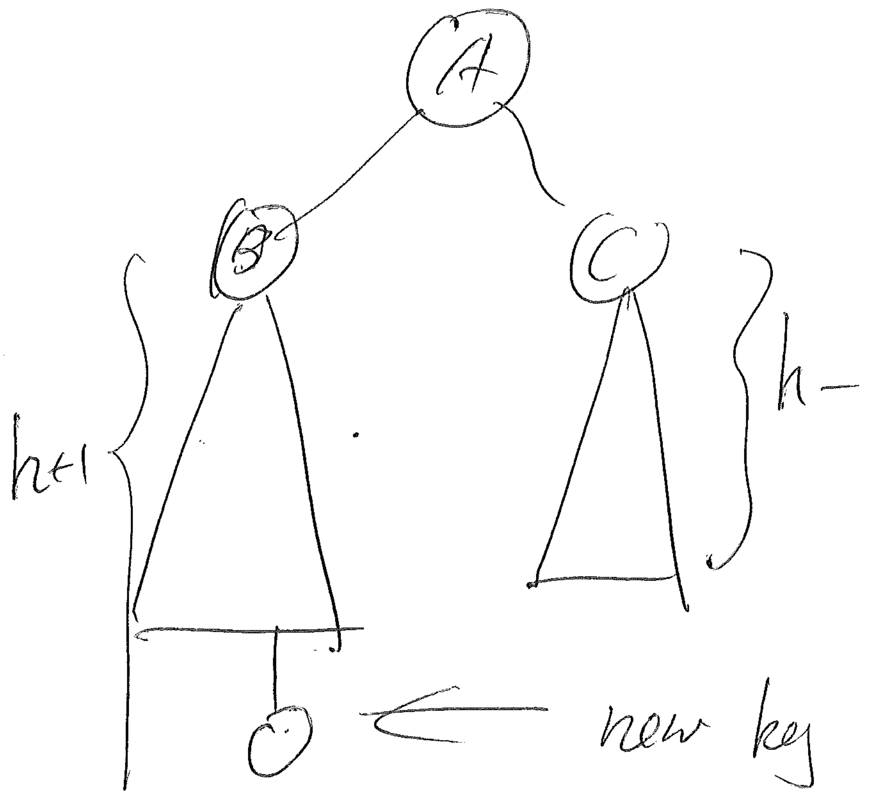
Let A be the ~~node~~ root of the smallest subtree that is violating the AVL property after insertion.





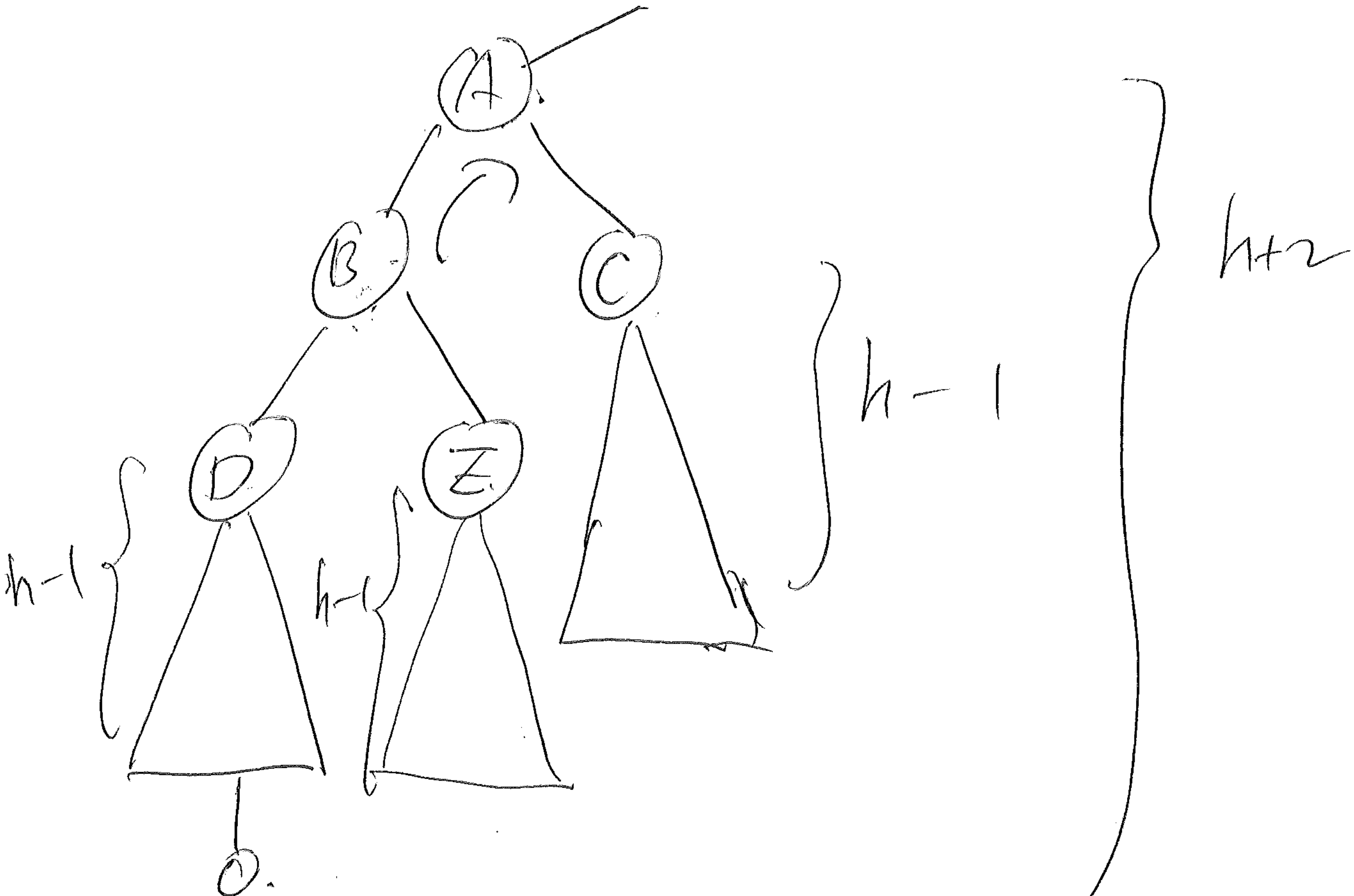
root of the smallest subtree violating AVL property

Assume the insertion happens in the subtree rooted at B.

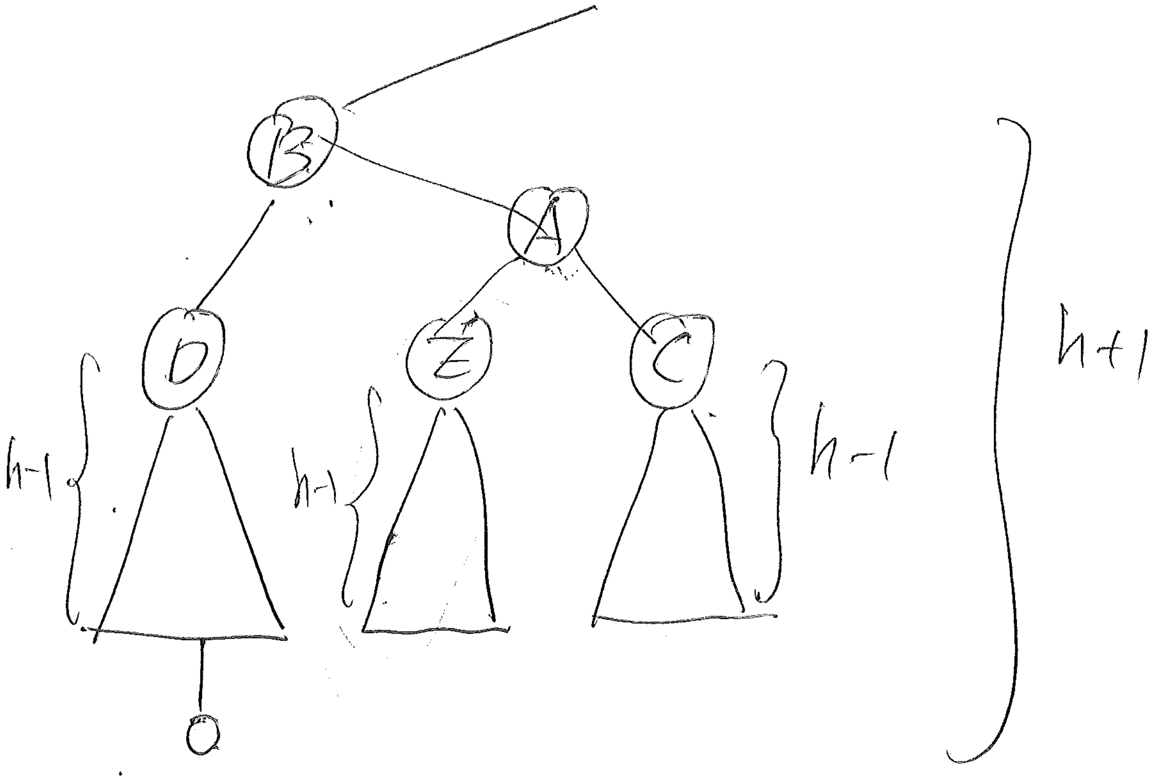


Two scenarios

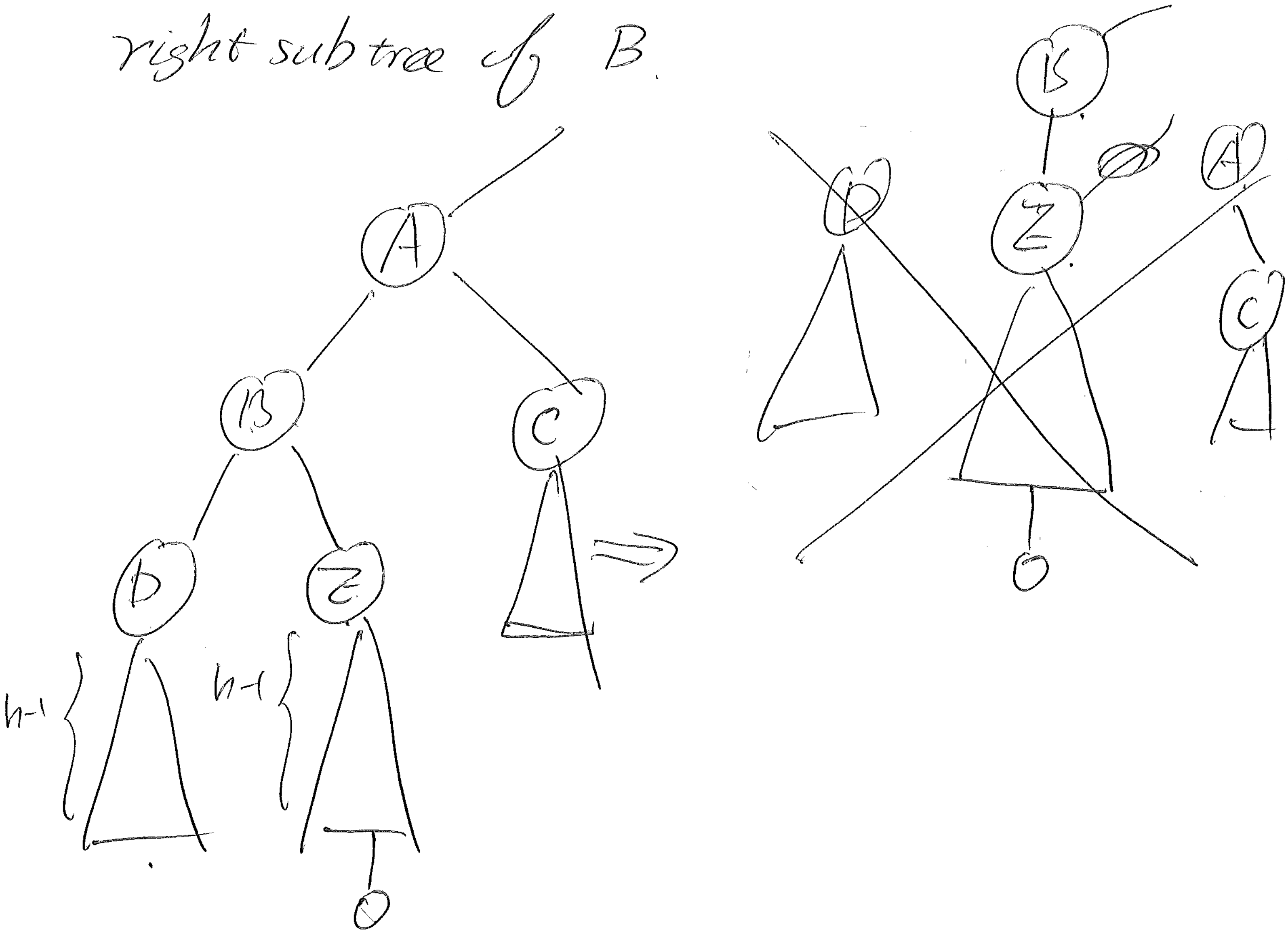
(1) The insertion happens to the left subtree of B

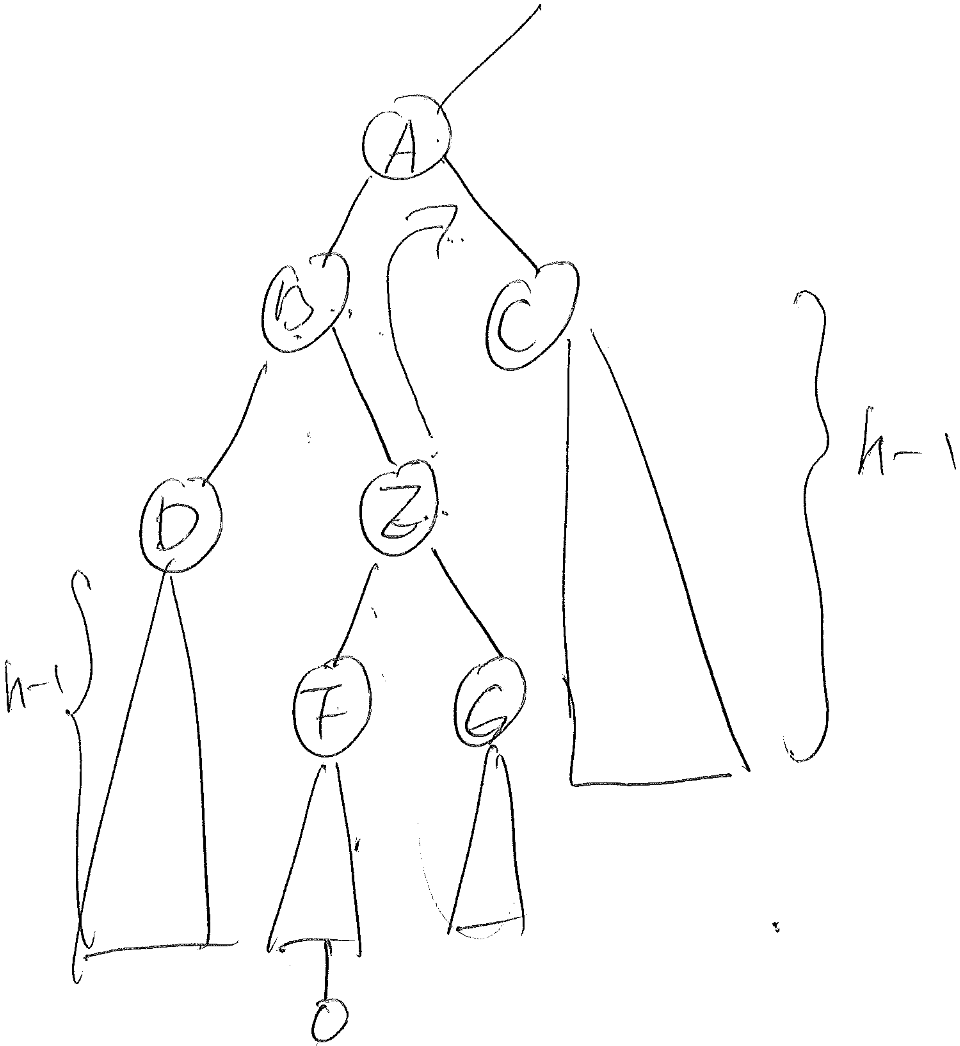


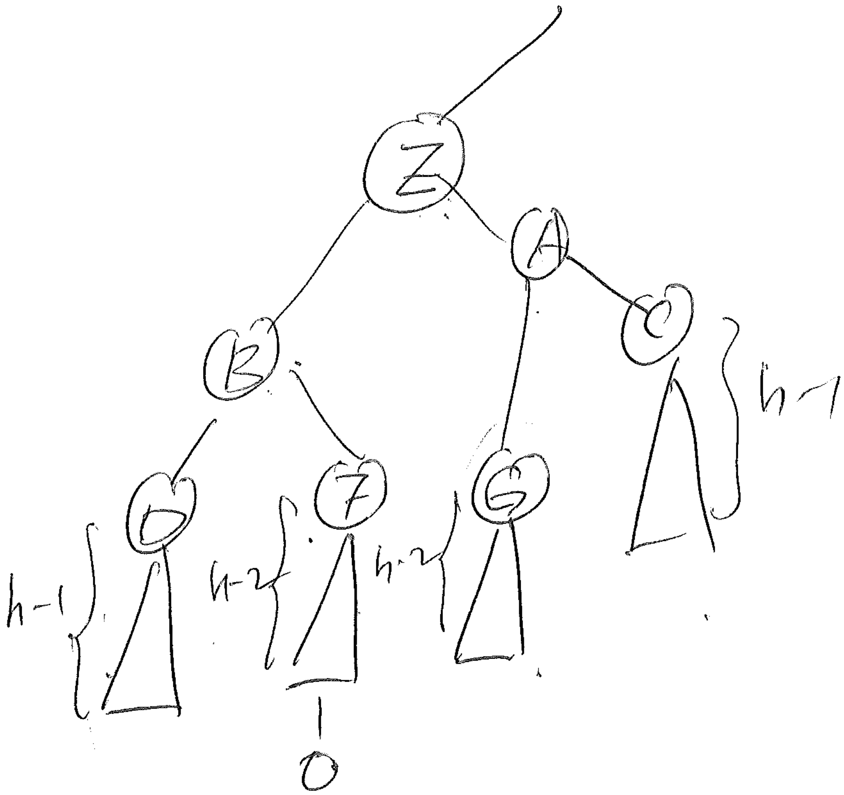
Strategy: Promote B to replace A.


B
A
D
Z
C
h-1
h-1
h-1
h+1

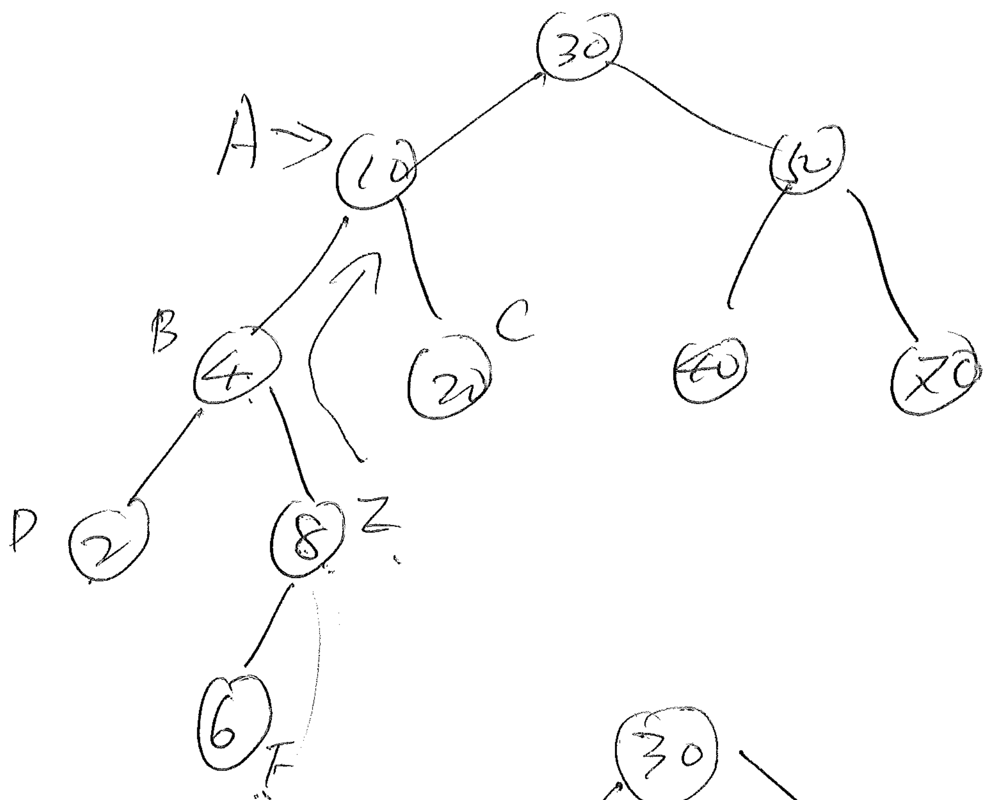
Scenario 2: insertion happens in the right subtree of B.

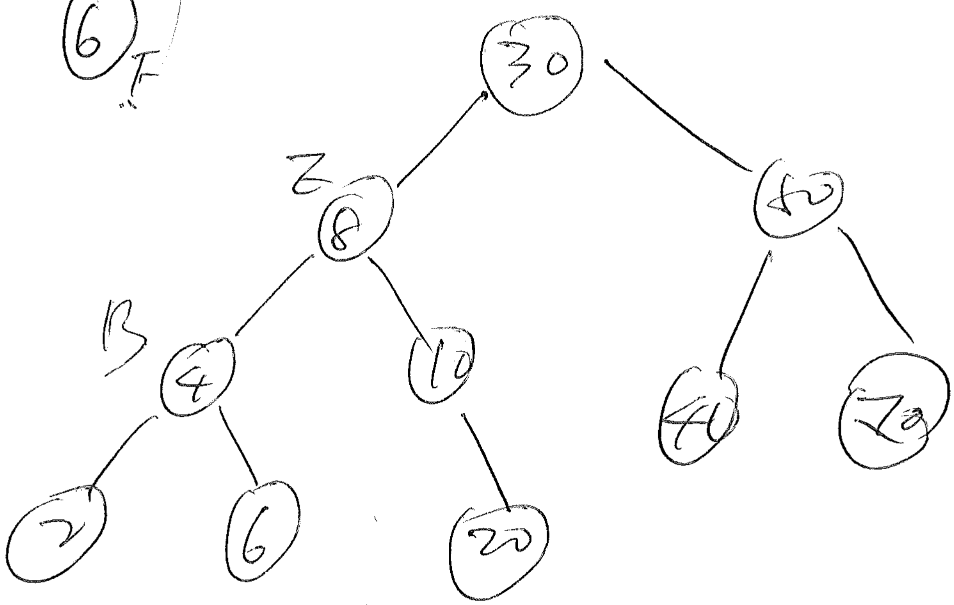



A
B
C
D
Z
F
G
h-1
h-1



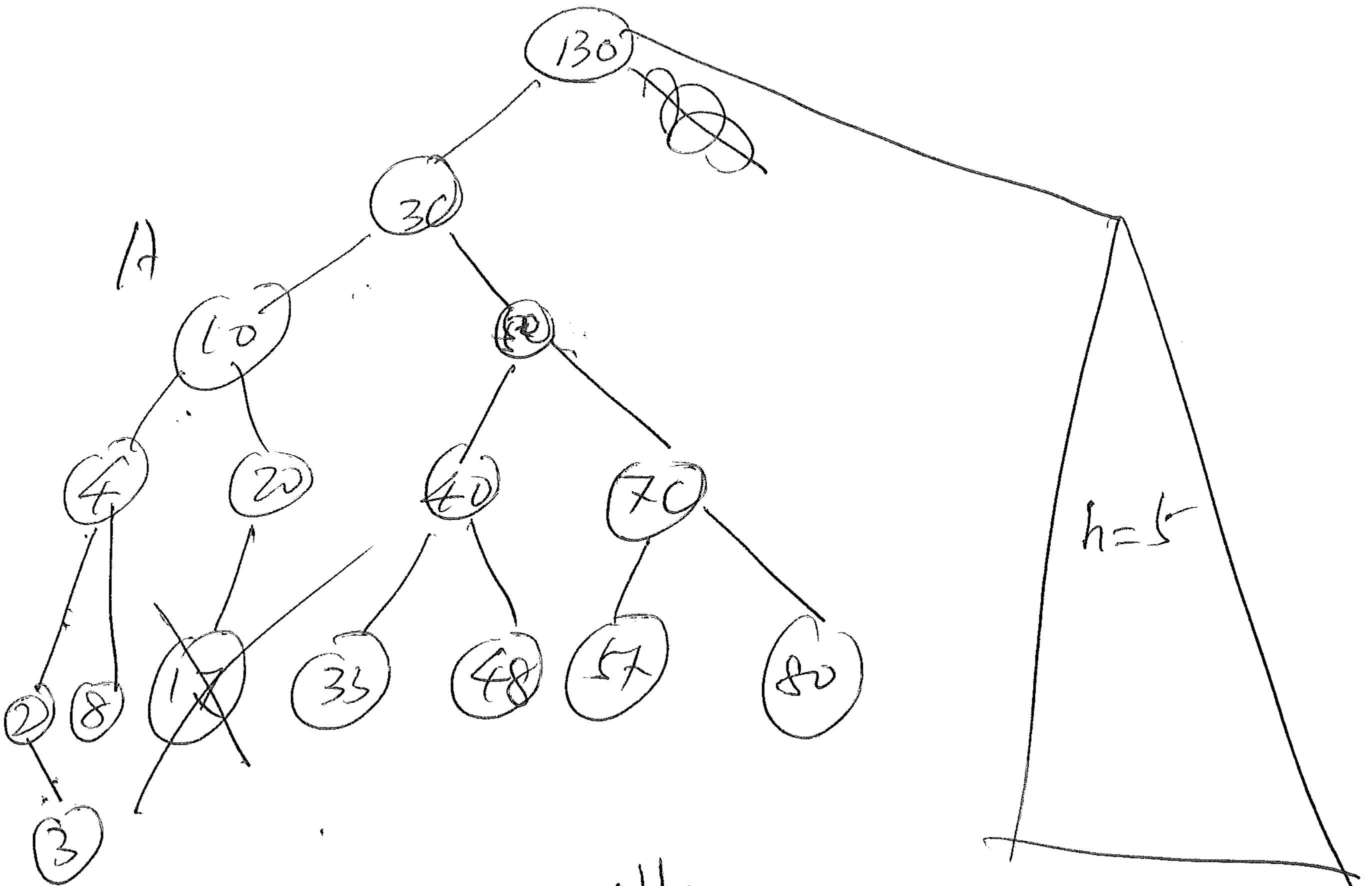
Double Rotatin

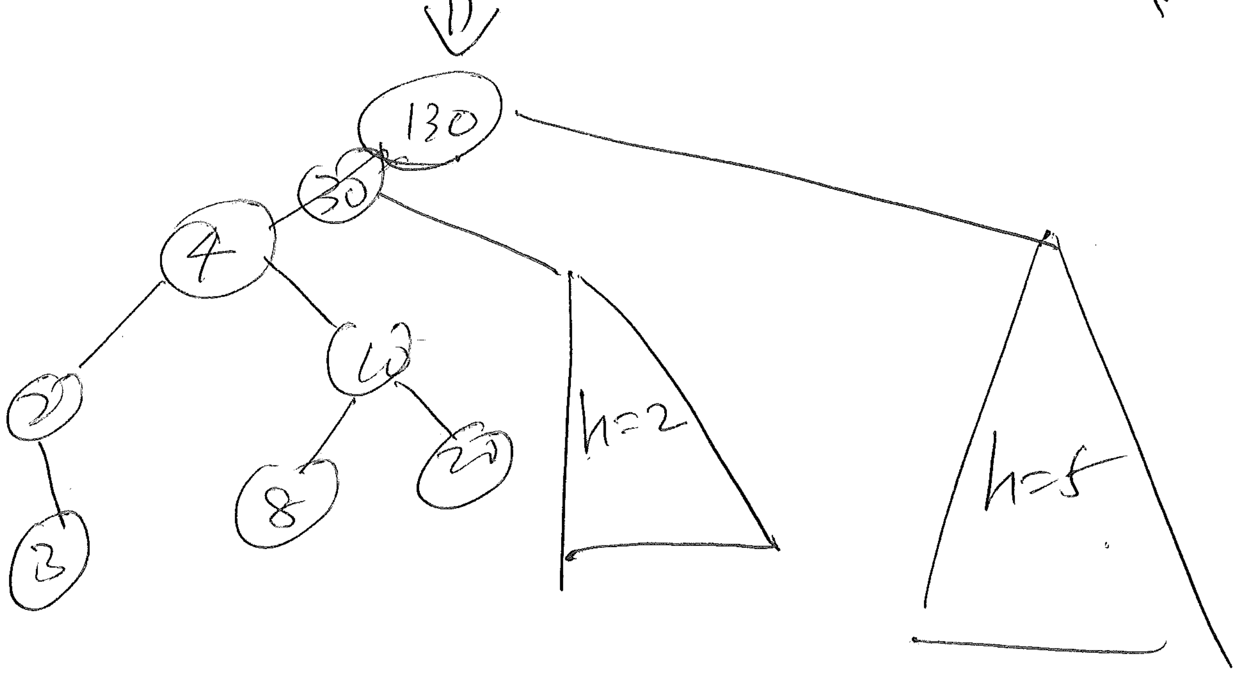

30
A
10
60
B
4
20
C
40
70
D
2
8
Z
6
F


30
Z
8
60
B
4
10
40
70
2
6
20

Deletion in AVL-tree

1. Normal deletion as in a BST
2. rebalance if necessary





Selection Algorithm.

Given an array $A[1..n]$ unsorted

and an integer k, $1 \le k \le n$

Find the k^{th} smallest number in A

Note when $k = \frac{n}{2}$ median selection

Naive Solution:

① Sort A ← $O(n \log n)$

② output $A[k]$

Reduce the running time to $O(n)$

Deterministic Linear Time Selection

① Partition A into groups of 5.

② Find the ~~medium~~ median of each group

③ Put all the medians into a set S'.

④ Recursively find the median x of S'

⑤ Use x as the pivot and partition A into

$S_1 \leq x < S_2$

⑥ If $k \leq |S_1|$, recursively ~~find~~ search S_1,

else recursively search S_2

A: 25 23 21 1, 5, 7, 15, 14, 11, 2

20, 17, 4, 18, 10, 11.5 12, 13, 3, 8

5.3, 15.1, 9, 1.8, 22

~~k=10~~ k = 7

$S' = \{21, 11, 17, 11.5\ 9\}$ $k' = 3$

$x = 11.5$

$S_1 = \{1, 5, 7, 11, 2, 4, 10, 11.5, 3, 8, 5.3$
$9, 1.8\} \leq x <$

$S_2 = \{25, 23, 21, 15, 14, 20, 17, 18, 12, 13$
$15.1, 22\}$

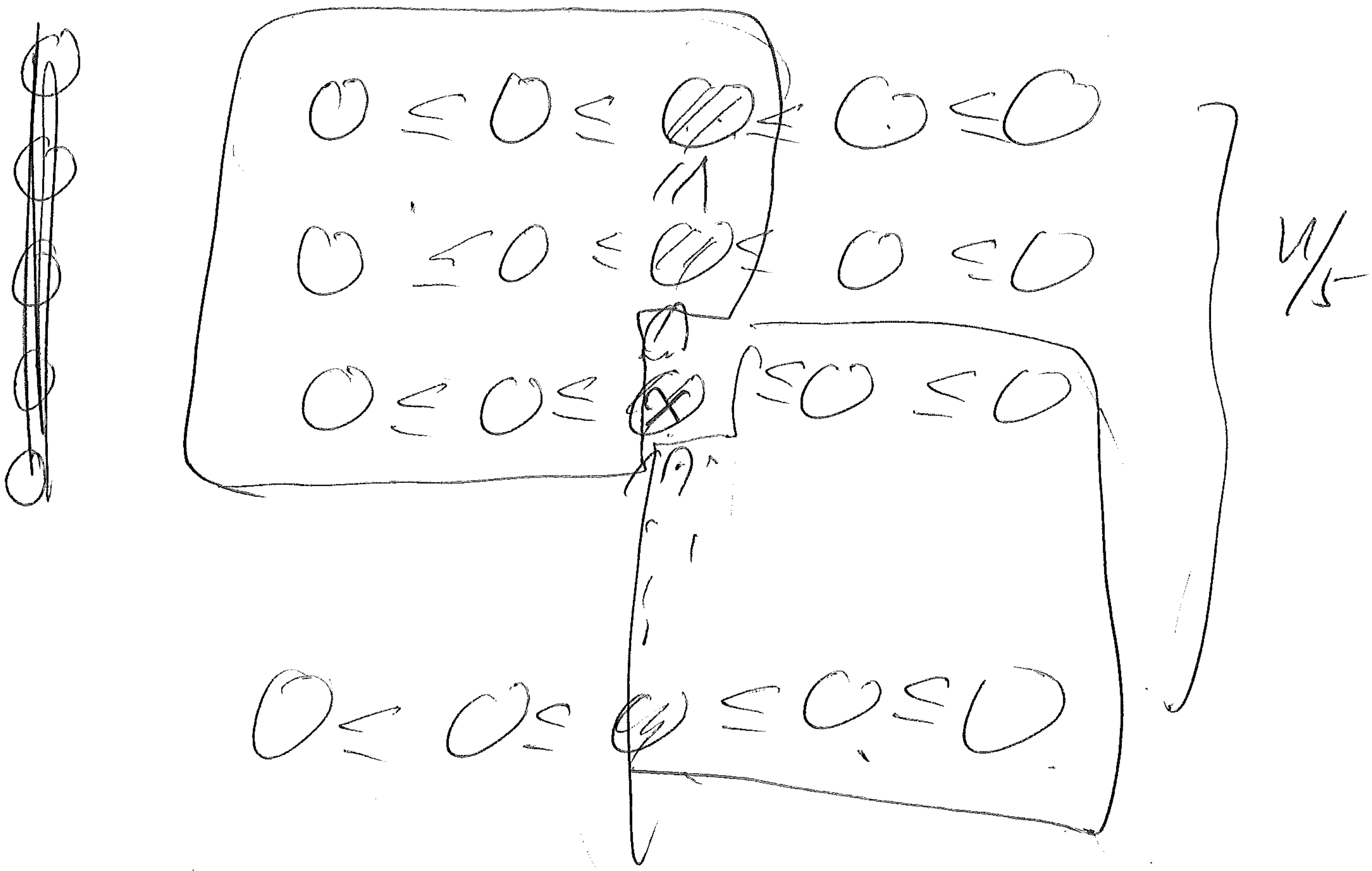
$|S_1| = 13$ $|S_2| = 12$

recursively search $(S_1, k = 7)$

if $k = 20$

recursively search $(S_2, k - |S_1|)$

$$T(n) = n + T\left(\left\lceil \frac{n}{5} \right\rceil\right) + T(\max\{|S_1|, |S_2|\})$$



$$|S_1| \ge \frac{n}{5} \cdot \frac{1}{2} \cdot 3 = \frac{3}{10} n$$

$$|S_2| \ge \frac{n}{5} \cdot \frac{1}{2} \cdot 3 = \frac{3}{10} n$$

$$|S_1| + |S_2| = n$$

$$|S_1|, |S_2| \le \frac{7n}{10}$$

$$T(n) = n + T\left(\frac{n}{5}\right) + T\left(\frac{7n}{10}\right)$$

$$\Rightarrow T(n) = \Theta(n) \approx 10n$$