COP 3502 - 10/30/23

Balanced Binary Search Trees - AVL Trees

In reg BST, insert, delete, search - O(h)
h = height. Best, avg case h = O(lg n), worst h = O(n).
n = # nodes.

Best known: **AVL Trees** Red-Black Trees, B-Trees

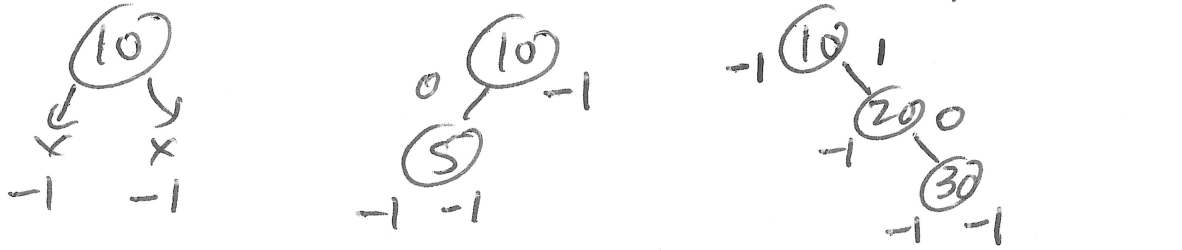
AVL Tree Def

Binary Search Tree s.t. each node satisfies the AVL Tree Node Property.

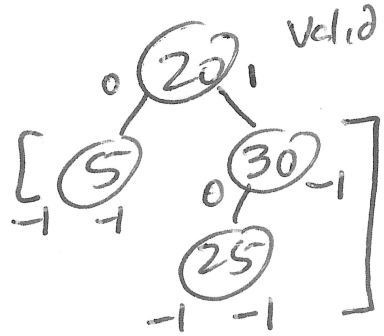
AVL Tree Node Property

for each node in the tree, the height of the left + right subtrees doesn't differ by more than 1.

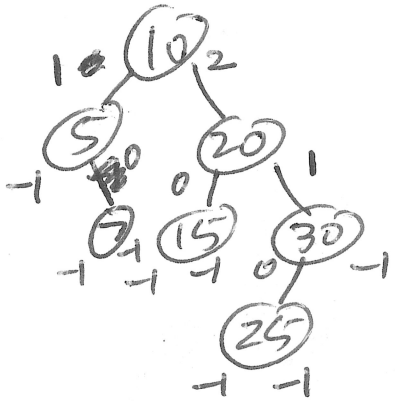
Not an AVL



Valid



Valid AVL h=2



Valid AVL
h=3
n=7

(1) Prove $h = O(\log n)$ for all AVL trees

Requires Mathematical Induction

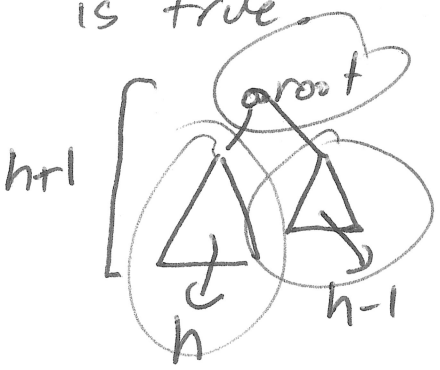
Prove that if an AVL tree has height h, then the ~~minimum~~ # of nodes it has is $F_{h+3} - 1$, where F_k represents the k^{th} Fibonacci #.

base cases: $h=0 \rightarrow n=1$ ✓ $\quad F_{0+3}-1 = F_3 - 1 = 2-1 = 1$ ✓

$h=1 \rightarrow n=2 \quad F_{1+3}-1 = F_4 - 1 = 3-1 = 2$ ✓

Inductive hypothesis: Assume for all non-neg integers $k \leq h$ that the given statement is true and $h \geq 1$.

Inductive step: Prove for $h+1$ that the statement is true.



One of the 2 subtrees must have height h. If we want to minimize # nodes, the other subtree should have height $h-1$, its min by AVL prop

$$\#\text{ nodes}_{\text{AVL tree height } h+1} \geq \left(\#\text{ nodes AVL height } h\right) + \left(\#\text{ nodes AVL height } h-1\right) + 1_{\text{root}}$$

$$= F_{h+3} - 1 + F_{h+2} - 1 + 1$$

$$= \boxed{F_{h+4} - 1}$$

$$n \geq F_{h+3} - 1$$

$$(n+1) \geq F_{h+3}$$

$$(n+1) \geq \frac{1}{\sqrt{5}} \left(\left(\frac{1+\sqrt{5}}{2} \right)^{h+3} - \left(\frac{1-\sqrt{5}}{2} \right)^{h+3} \right)$$

$\frac{1+\sqrt{5}}{2} \to \phi \sim 1.6$ golden ratio

$$\log_\phi (n+1) \geq h+3 \quad \text{(approx)}$$

$$h \leq \log_\phi (n+1) - 3 = O(\lg n)$$

Insert

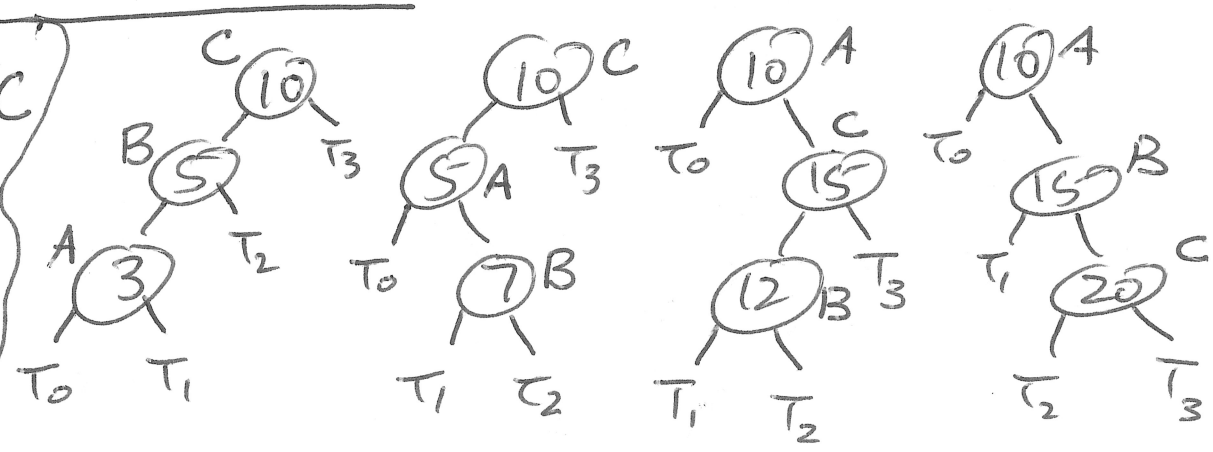
$A < B < C$

$T_0 < A$

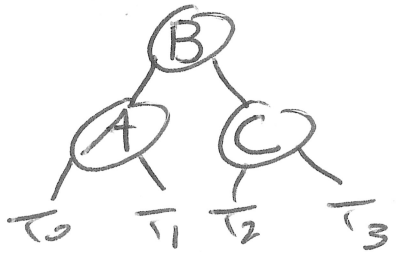
$A < T_1 < B$

$B < T_2 < C$

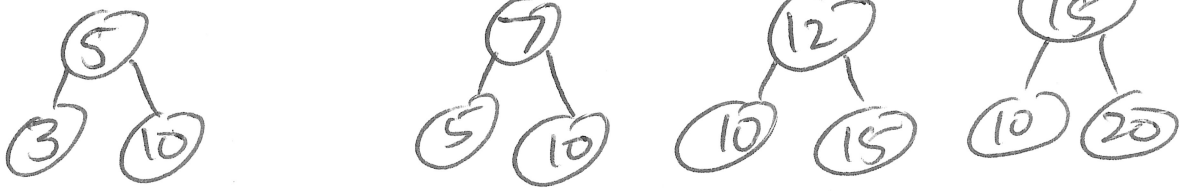
$T_3 > C$



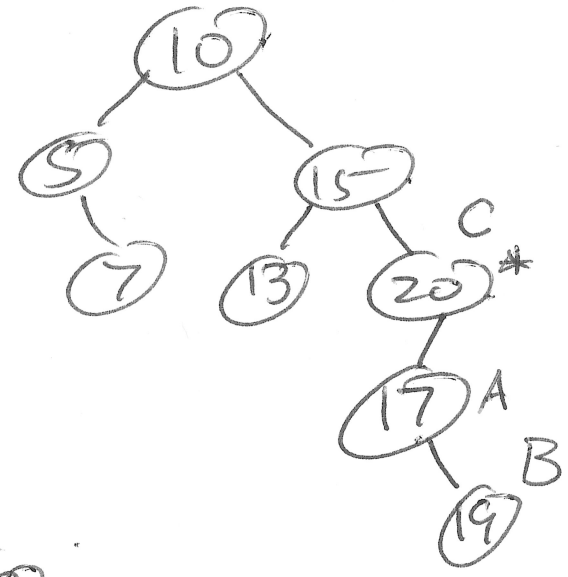
Fix in all cases:



fixes

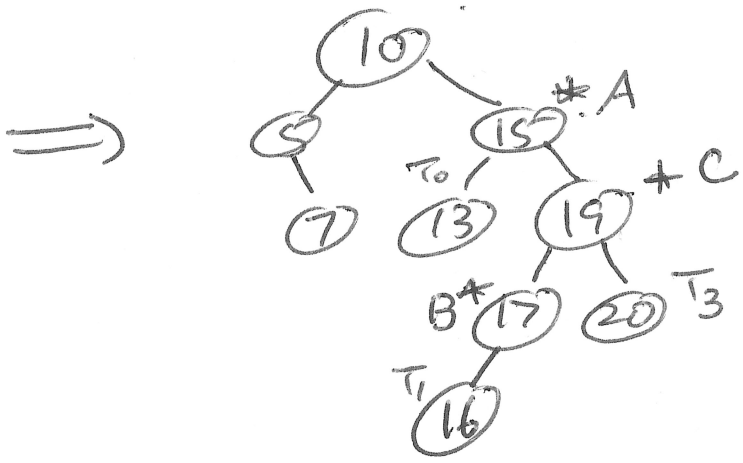


Examples

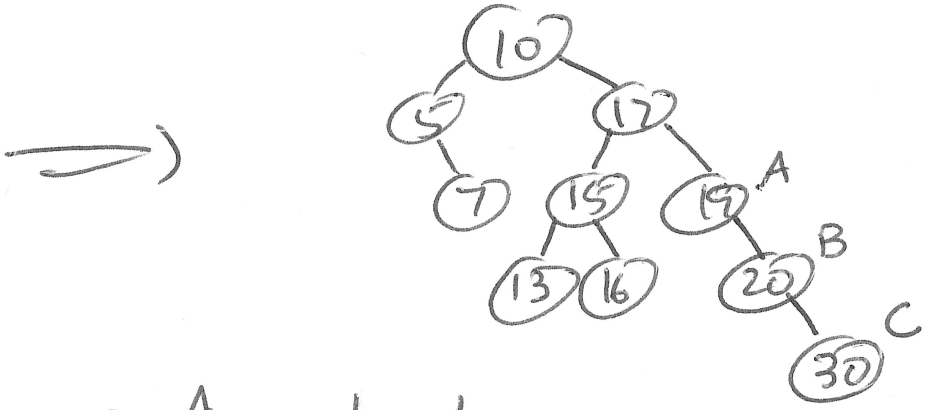


Insert 19

1. Reg BST insert
2. Trace back ancestral path + check if a node is imbalanced
3. If so fix it



Insert 16



Insert 30