Maximum number nodes at level l in a binary tree (levels are 0-indexed):

2^l
$2^{l+1} - 1$
$2^{l-1} + 1$
$2^l + 1$
$2^l - 1$

Maximum number of nodes in binary tree with height h:

2^h
$2^{h+1} - 1$
$2^{h-1} + 1$
$2^h + 1$
$2^h - 1$

$2^0 + 2^1 + \ldots + 2^h = 2^{h+1} - 1$

A binary tree has N nodes. Then height of the tree (choose the most strict one):

$h \geq \lg N$
$h > \lg N$
$h \geq \lg N + 1$
$h \geq \lg N - 1$
$h \geq \lg(N+1)$
$h > \lg(N+1)$
✓ $h \geq \lg(N+1) - 1$
$h > \lg(N+1) - 1$
$h \leq 2\lg(N+1)$
$h \leq \lg N + 1$

$N \leq 2^{h+1} - 1 \Rightarrow N+1 \leq 2^{h+1}$

$\Rightarrow \lg(N+1) \leq h+1$

$\Rightarrow h \geq \lg(N+1) - 1$

_____ traversal of a BST is the elements in sorted order.

- Pre-order
- In-order
- Post-order
- None. Because otherwise we would have $O(n)$ sorting which is impossible.

Complexity of

```
for (auto it = s.begin(); it != s.end(); ++it)
    cout << *it << endl;
```

$O(N)$ ≈ In-order trav.

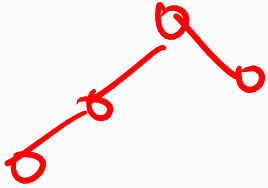
$O(N \lg N)$

Complexity of

```
for (int i = 0; i < N; ++i){ {
auto it = median;
cout << *(++it) << endl; }
```

- $O(N)$
- ✓ $O(N \lg N)$

An AVL tree has height 2. Minimum number of nodes it may have: 4



An AVL tree has height 6. Minimum number of nodes it may have: 33

use $N_n = N_{n-1} + N_{n-2} + 1$

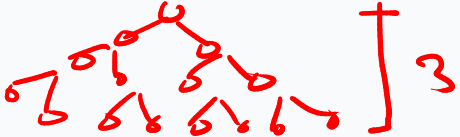
Maximum number of rotations for an insert in an AVL tree with N nodes and h height is (assume LR, RL is two rotations): 2

Minimum number of nodes required to get X rotations for an delete operation is:

- N_X
- N_{X+1}
- N_{2X}
- N_{2X+1}

Minimum height of an AVL tree with 15 nodes: 3

as compact as possible:



Maximum height of an AVL tree with 15 nodes: 4

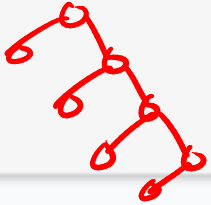
$N_4 = 12$, $N_5 = 20$, Need 20 nodes for $h=5$.

so max is 4.

If every node of a binary tree has 0 or 2 children, then height is $O(\lg N)$.

True

False



$h = N/2 = O(N)$

In an AVL tree, the median is either the root node, or one of its two children.

True

✓ False

only height balanced, not size. Look at N_6.

N inserts in an AVL tree can be $O(N)$.

True

this is Σ min depth. for each insert.

min depth $\geq \frac{\text{maxdepth}}{2}$

[convince yourself from $N_2, \ldots, N_6$]

False

so, $\Sigma \frac{\log(i)}{2} \approx O(N \lg N)$.

Insert N elements into an AVL tree. Total number of rotations is:

$O(1)$
$O(\lg N)$ → possibly??
✓ $O(N)$ each take 2 rotation at most
$O(N \lg \lg N)$
$O(N \lg N)$ Can you find a sequence??

At any node in an AVL tree, left and right subtree differ by ≤ 1 height. As a result, any two leaf node differ by ≤ 1 depth.

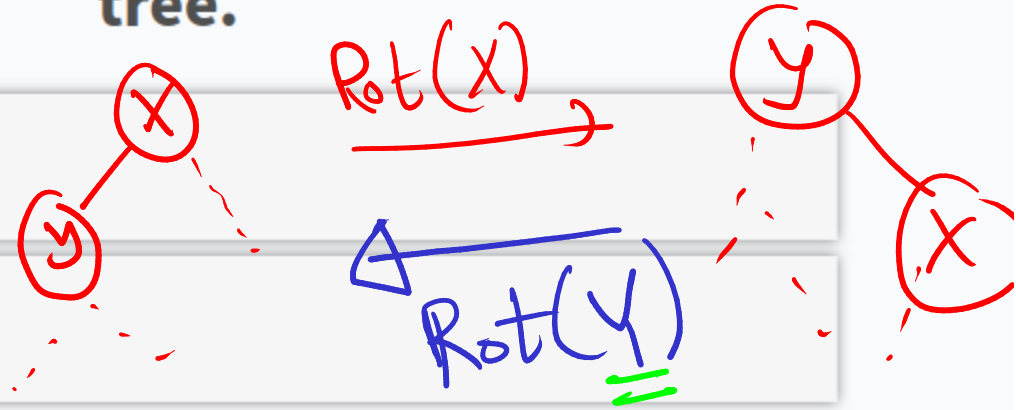
- True
- ✓ False

look at N_6

Let X be any node in an AVL tree. $\mathrm{LeftRotate}(X)$ followed by $\mathrm{RightRotate}(X)$ does not change the tree.

True

False



AVL tree with height h has at least 2^h nodes.

True

False

$N_3 = 7 \not> 8$

After an insertion, before doing any rotations the maximum imbalance of a node is:

- 1
- 2
- 3
- h