Today:

Monday, March 5, 2018 9:58 AM

- Tree Complexity
  - balanced vs Unbalanced

Tree Balancing

Red-Black Trees

Complexity

- Balanced vs. unbalanced

Number of tree nodes:

$N = 2^0 + 2^1 + 2^2 + \ldots + 2^h$

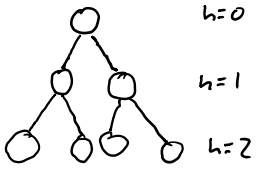
$N = 2^{h+1} - 1$

$\log_2(N+1) = \log_2(2^{h+1})$

$\log_2(N+1) = h+1$

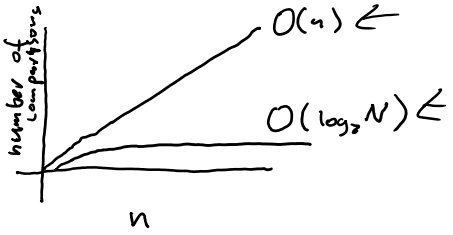
$h = \log_2(N+1) - 1$

$h \approx O(\log_2 N)$



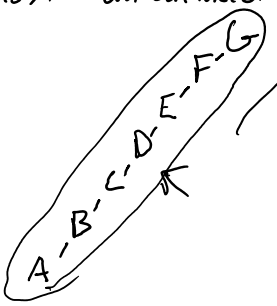
$2^3 - 1 = 7$

$O(\log_2 N)$ is much better scaling than $O(n)$.



However, will only get this relationship if the tree is balanced.

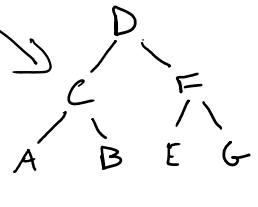
The most unbalanced tree we can have:



$h = N - 1$

$O(N) = O(7)$

$O(N) = 7$



h=2

$\log_2(N) = \log_2(7) = 2.8$

$O(\log_2 N) = \log_2(1000000) = 19.9$

$O(N) = 1e6$

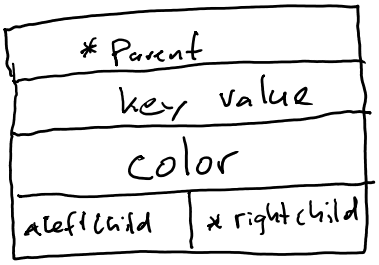
Tree Balancing

Various algorithms, e.g. AVL, red-black.

Work by having nodes w/ additional information (properties) and performing rotations on the trees in a way that preserves the BST definition.

Red-Black Trees

Special case of BST



Like standard BST, except extra "color" property added.

color - red or black

RB Tree Definition

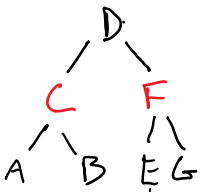
Property 1: A node is either red or black

P2: Root node is black.

P3: Every leaf node (node w/ null children) is black.

P4: If a node is red, both its children must be black.

P5: For each node, all paths to leaf nodes contain same number of black nodes.

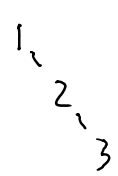


RB Tree Operations

In order for tree to self balance, certain operations are needed.

Recolor - change node color
red → black black → red

Rotate - changes height of (sub) tree
- change which node is root



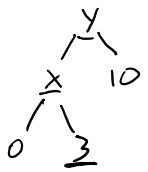
rotate left

1 is root
2 is right child of root
h = 2

2 is root
1 is LC of root
h = 1

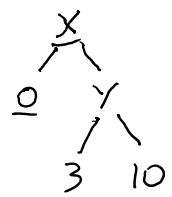
Left and Right Rotations

2 possible rotations. One is inverse of the other.
- Rotate Right alters tree. Rotate left gets back to original tree.



y is root
$x > 0$
$x \leq 3 < y$
$y \leq 10$

right rotate



x is now root
$y > x$, so y is now RC of x
$x \leq 3 < y$
3 assigned as LC of y.

$x > 0$, $y \leq 10$

Inserting a Node into a RB Tree

Same as inserting a node into regular BST, w/ a few additional steps.

1. Instead of assigning nullptr in 1 or 0 child scenarios, add Null (empty) sentinel node.
2. Set color of new node to red.
3. Resolve any RB property violations by using recoloring and/or rotations.

Examples: Building a RB tree

insert (10) into empty tree:

Rule: new node is red

Violation: root of tree must be black

Fix: re-color node to be black

Add 5 to tree

New node is red

No violations