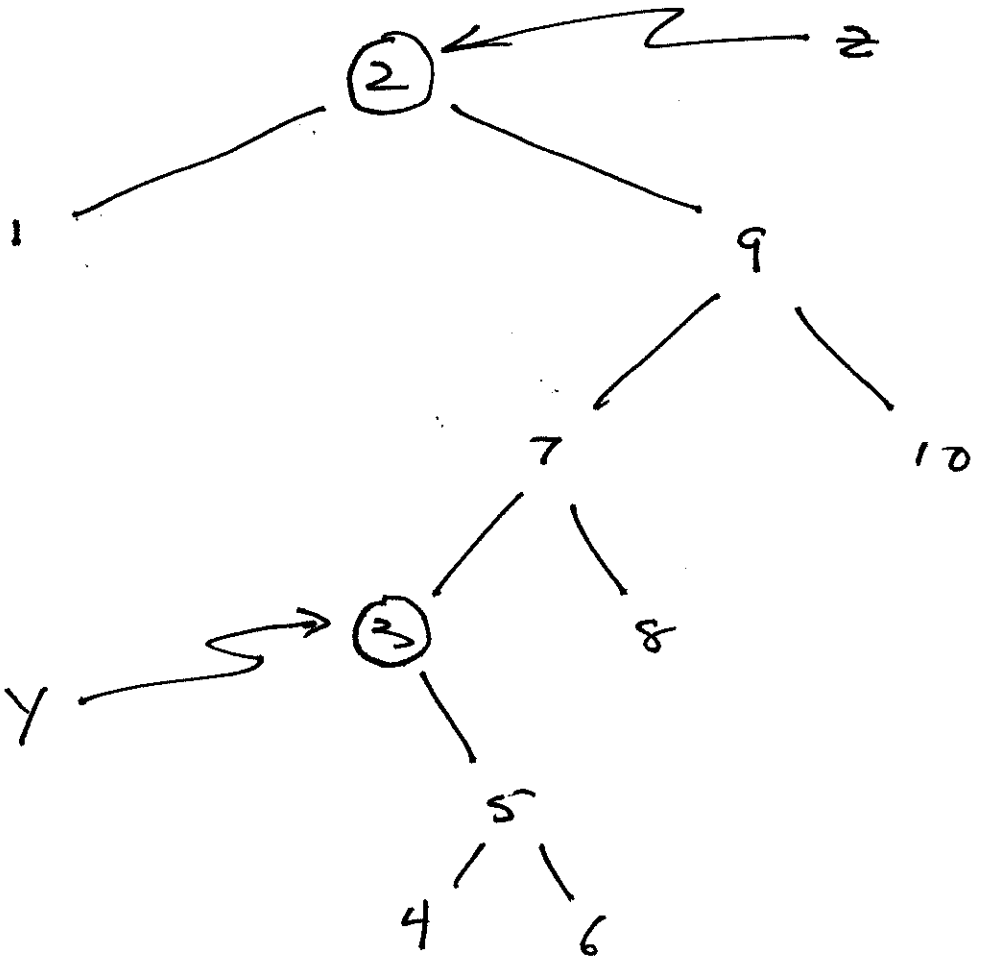
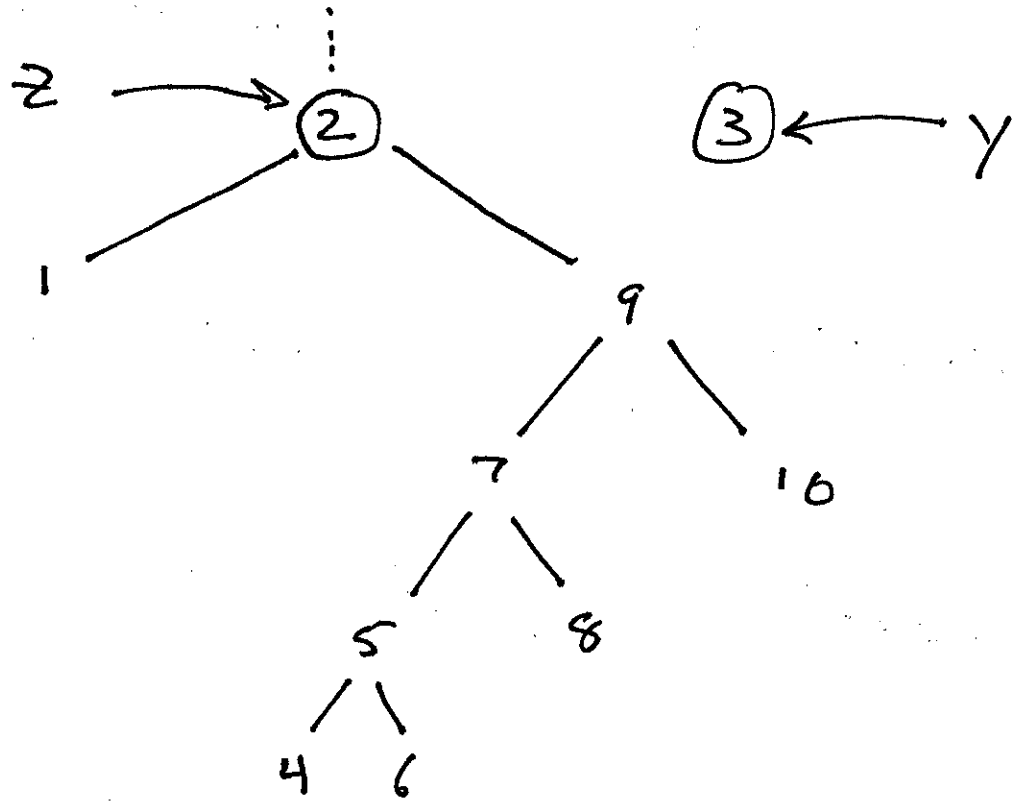
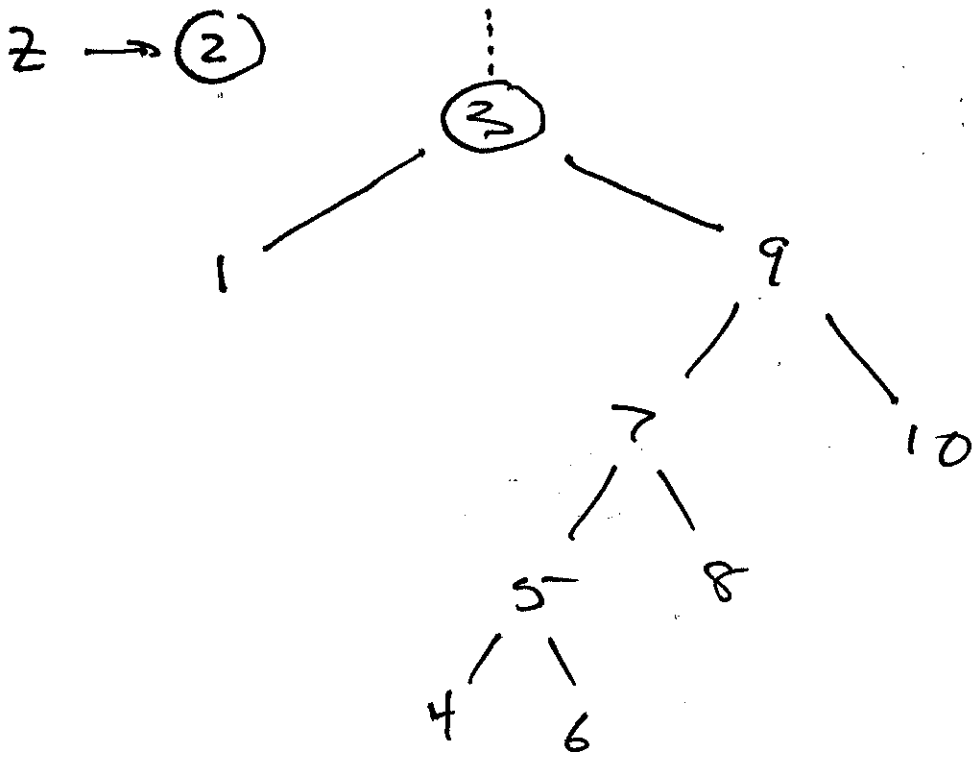
CSE 101 - 01 5-30-23

Pa7 : ext. 1 day → Thur.

Ex. Delete(T, 2) (case 3)

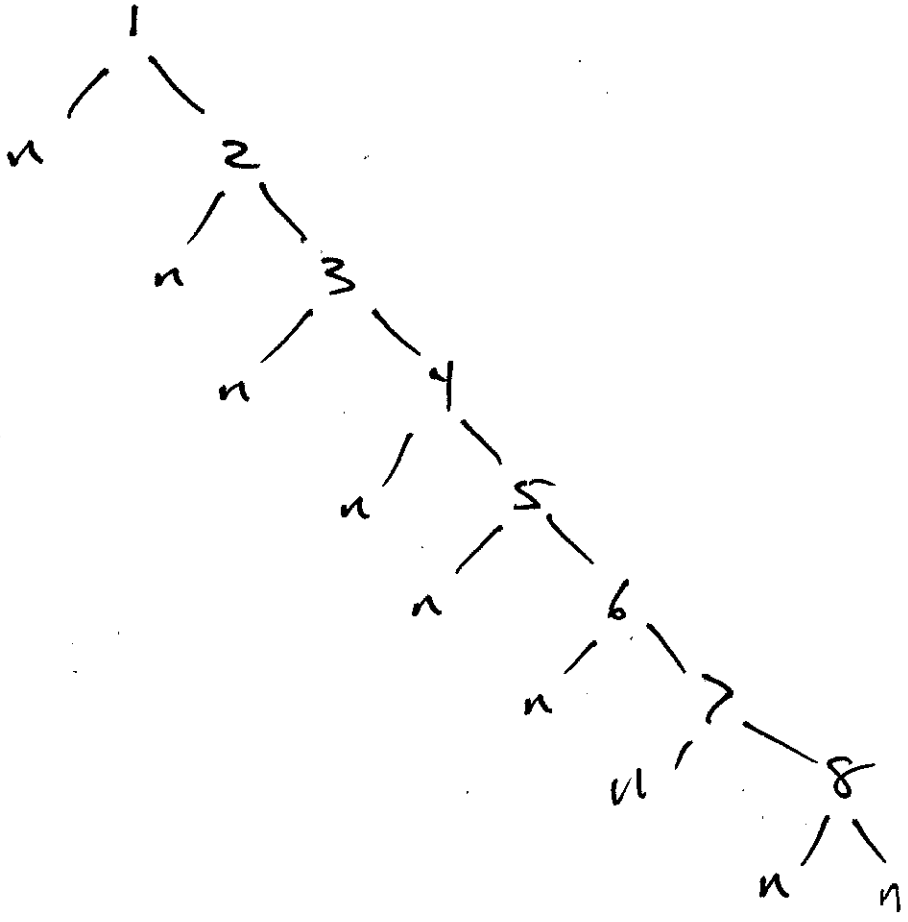



z
2
3
y
1
9
7
10
5
8
4
6


z
2
3
1
9
7
10
5
8
4
6

Problem with BST's:

Ex. insert: 1, 2, 3, 4, 5, 6, 7, 8



Recall: runtime of query, insert, delete all run in time

$O(\text{height}(T))$.

Solution Red-Black Trees (RBT)

(chap. 13 in CLRS)

Def: A RBT is a BST satisfying the RBT properties. Convention: the leaves in a RBT are the nil children of key-bearing nodes.

RBT Properties:

1.) Each node has a color: Red or Black.

2.) The root is Black

3.) Each leaf (i.e. nils) is Black.

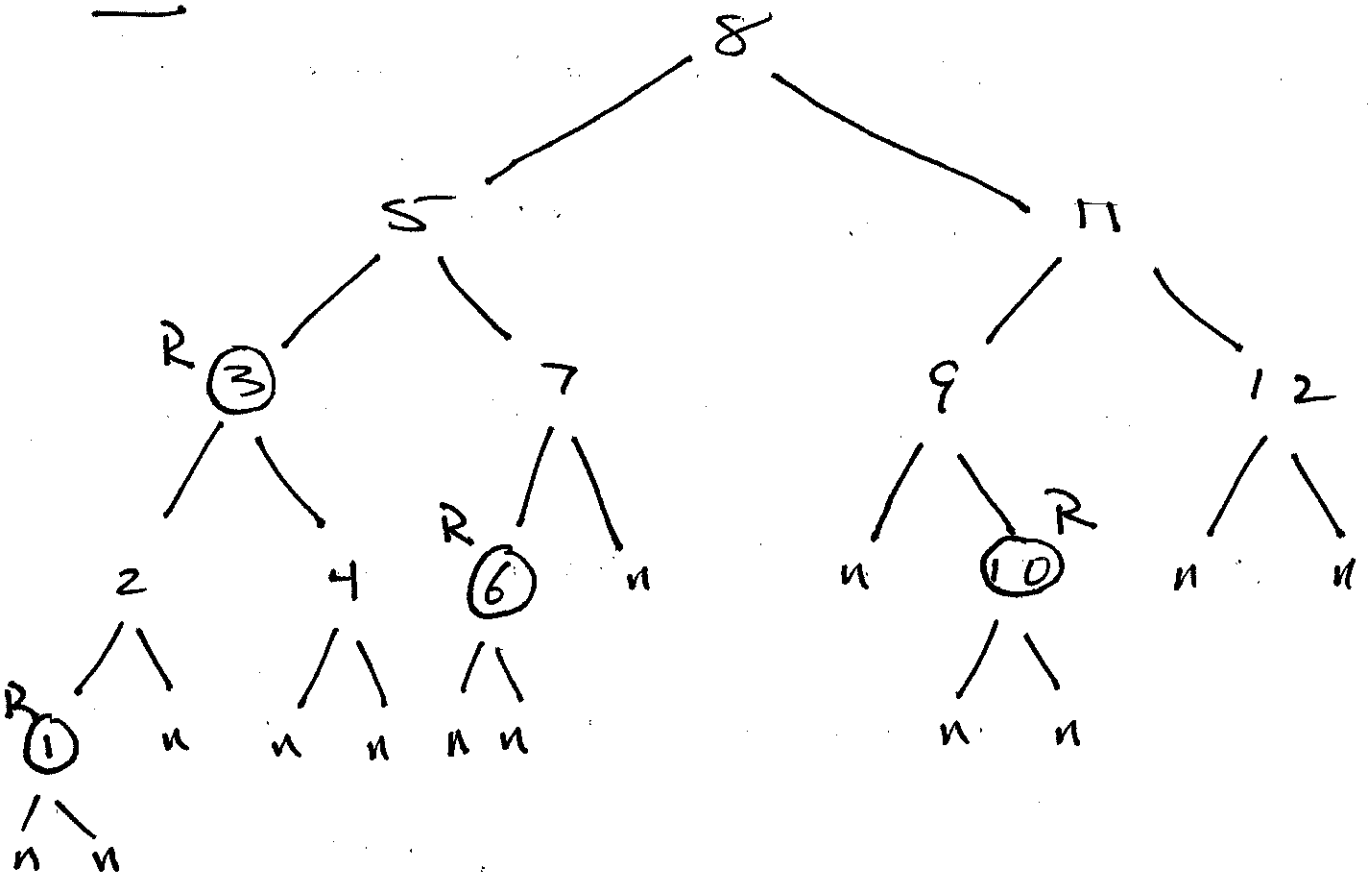
4.) Every Red node has 2 black children (Possibly nil.)

5.) For any node x, every descending Path from x to a leaf has the same number of Black nodes.

Defn

The Black-height of node x, bh(x) is the # of black nodes in any desc. Path from x to a leaf, not counting x.

Ex.



Exercise: are there other distribution of Red nodes that would make this BST satisfy RBT Properties.

note: $bh(x) = 0$ iff $height(x) = 0$

iff x is a leaf

Theorem

A RBT with n internal (i.e. non-nil) nodes and height h satisfies: $h \leq 2 \lg(n+1)$.

Remarks

- any Binary Tree satisfies $h \geq \lfloor \lg n \rfloor$, so a RBT satisfies

$$\lfloor \lg n \rfloor \leq h \leq 2 \lg(n+1)$$

so $\Omega(\log n) \leq \text{height}(T) \leq O(\log n)$

$\therefore$ $\text{height}(T) = \Theta(\log n)$

- any tree satisfying

$$\text{height}(T) = \Theta(\log n)$$

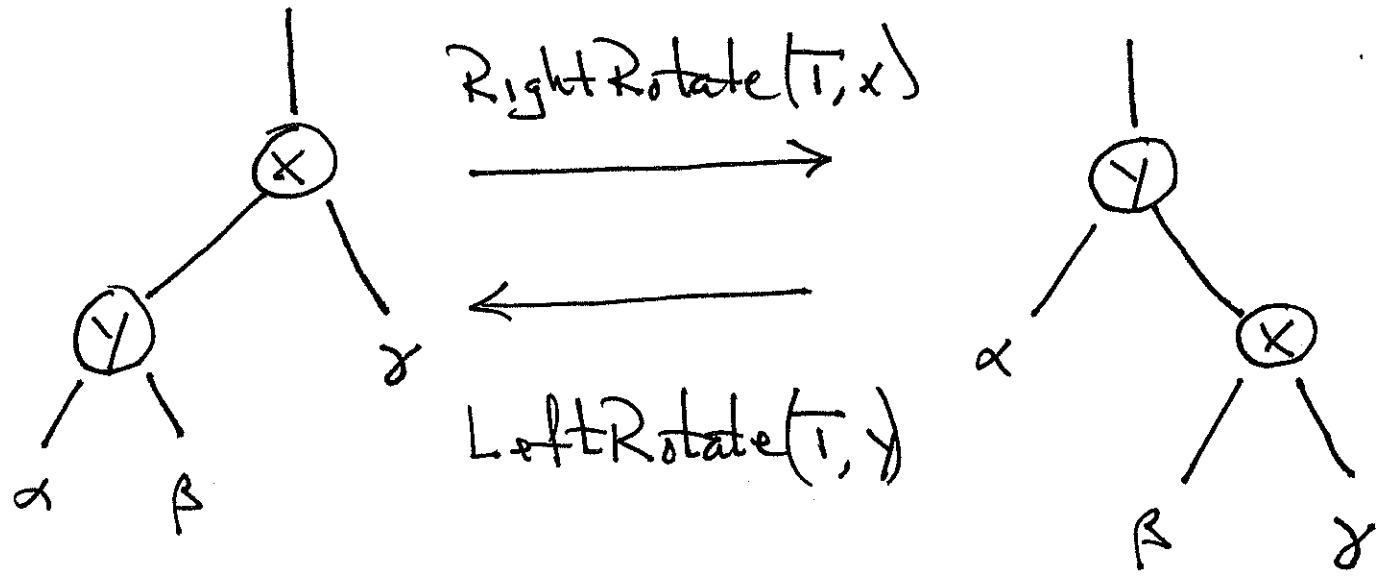
is called Balanced.

- Insert, Delete in a BST do not preserve RBT properties

- Insert & Delete can be altered so as to preserve RBT properties, and run in time

$$\Theta(\text{height}(T)) = \Theta(\log n)$$

13.2 Rotations



Summary: