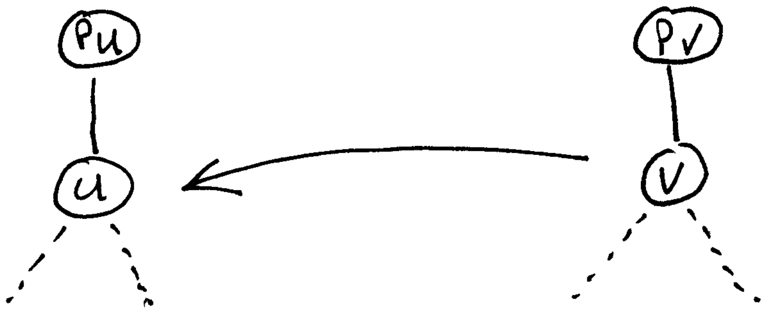
cse 101 2-16-21

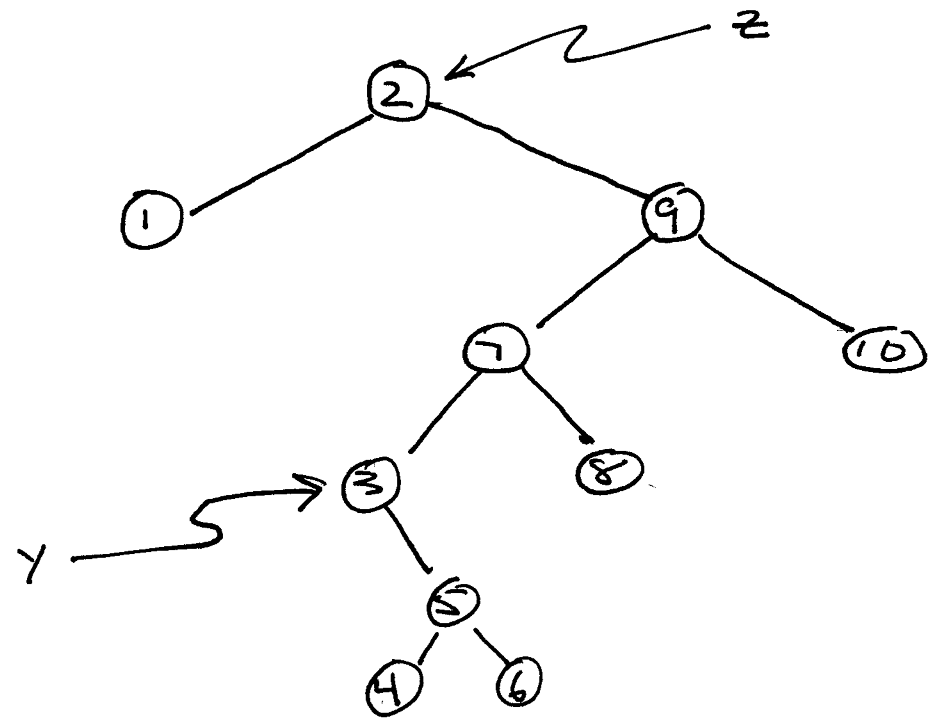
PA5: ext. 2 days (last)

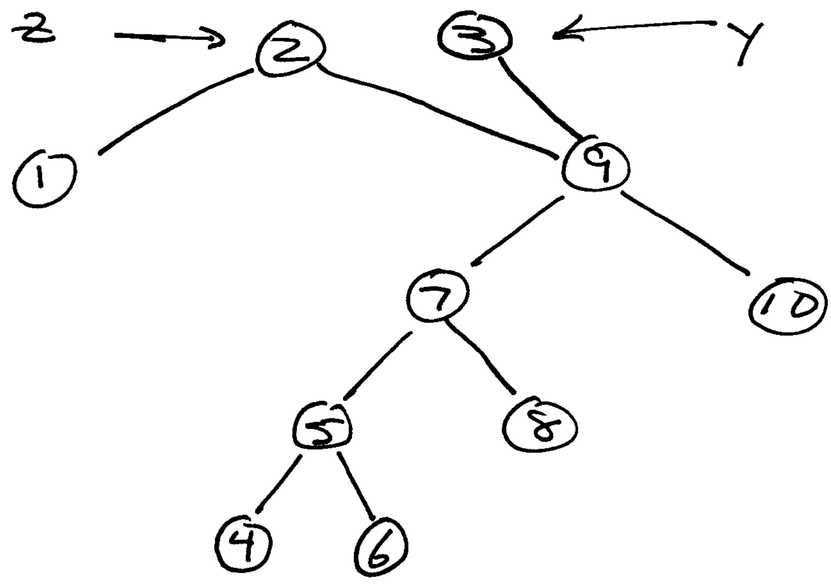
Continue BST delete: Case 3.

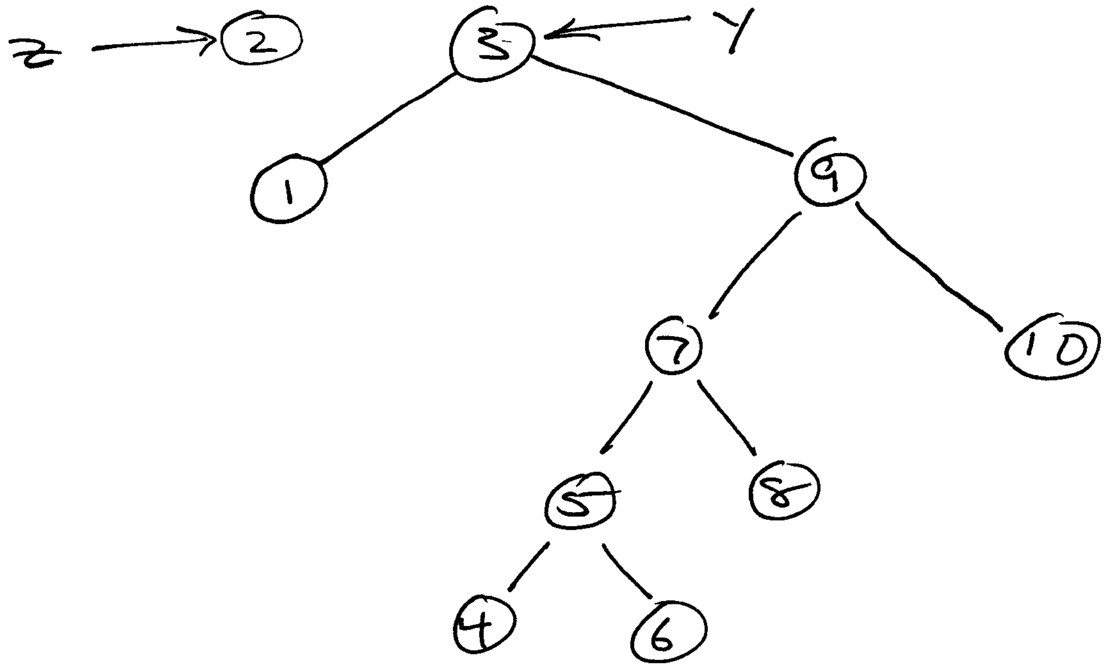
Helper function: Transplant(T, u, v)



Ex (case 3)






done!

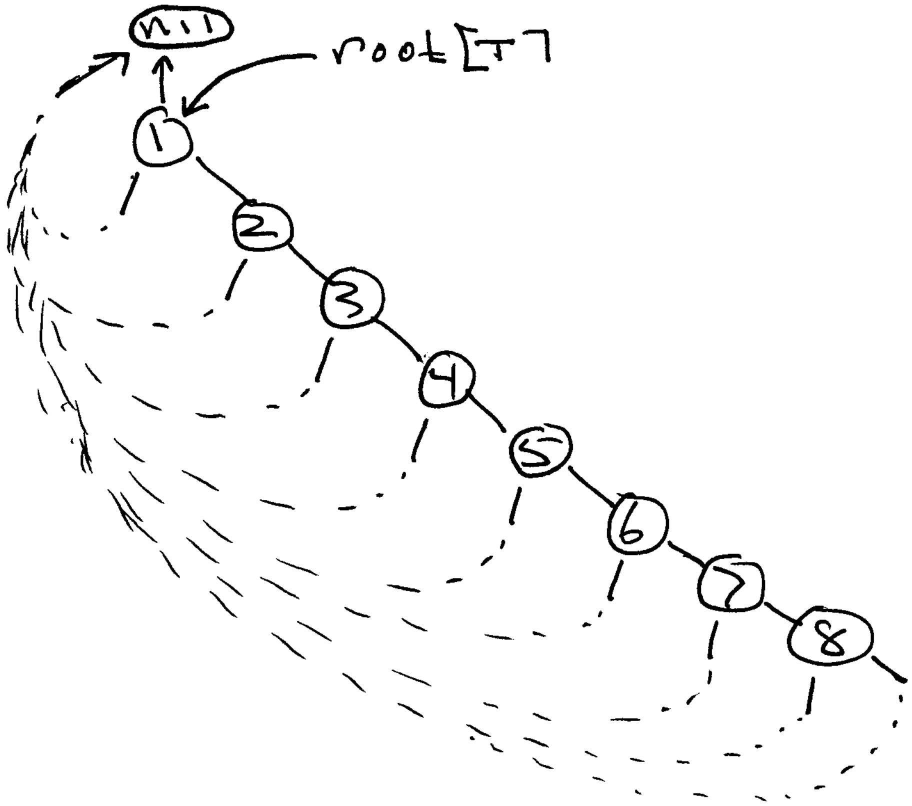
Pa5 stuff . .

forward & backward iterations

| Dictionary — (analogous to) — | List (Pa1) |
| --- | --- |
| beginForward() | moveFront() |
| beginReverse() | moveBack() |
| currentKey() <br> currentVal() | get() |
| next() | moveNext() |
| prev() | movePrev() |

Recall: if we insert keys:

1 2 3 4 5 6 7 8 in order, we get




Runtime of insert(), lookup(), delete()

are all $\Theta(\underbrace{\text{height}(T)}_{n})$, in worst case.

# chap. 13 Red-Black Trees (RBTs)

Goal: Balance a BST so that insert(), lookup(), delete() are efficient (in worst case.)

Defn.
A RBT is a BST that satisfies the RBT properties (in addition to the BST properties.)

convention: the leaves in a RBT are considered to be the nil children of key-bearing nodes

# RBT Properties.

1. each node has a color attribute Red or Black.

2. root[T] is Black.

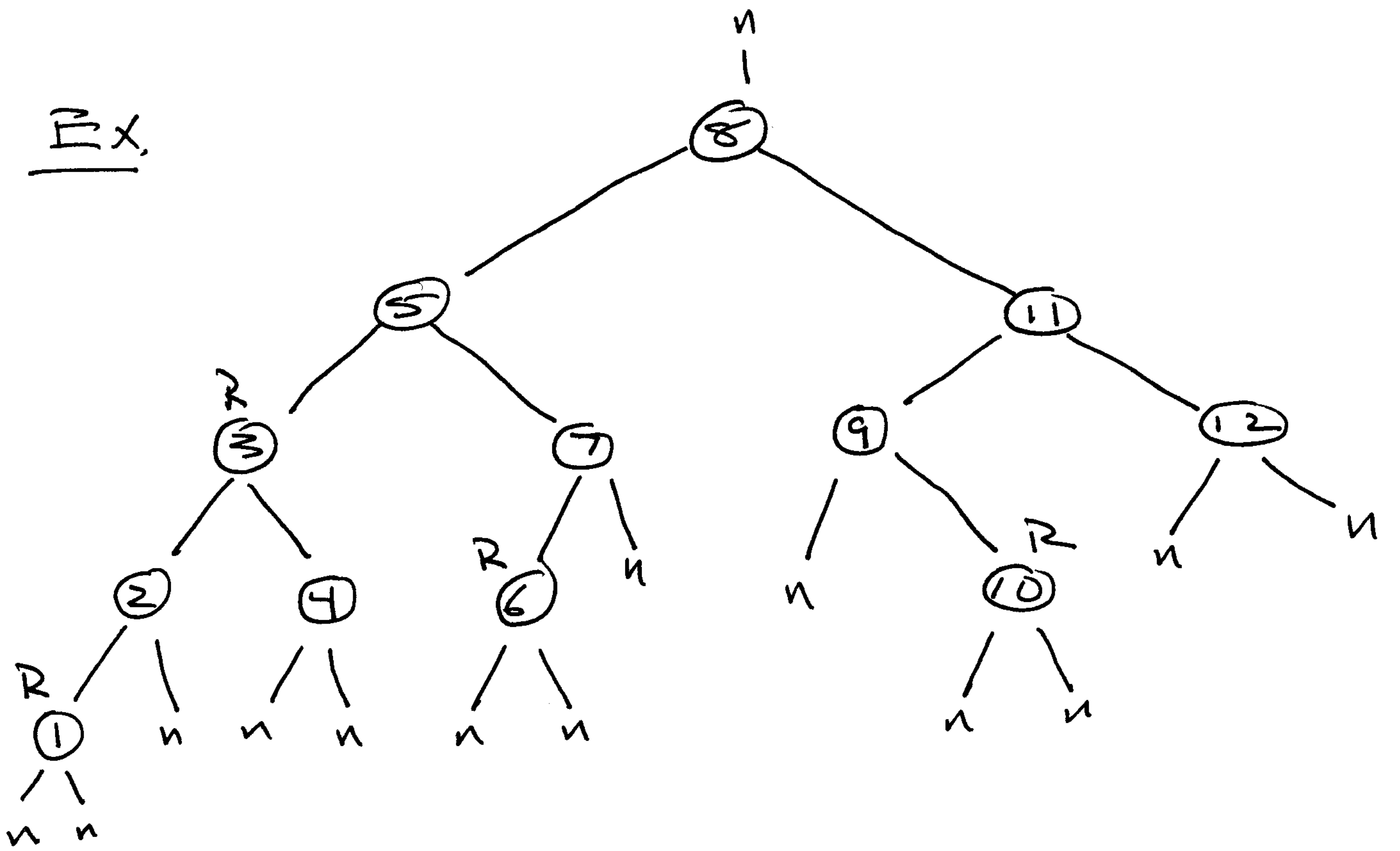
3. each leaf (nil child) is Black.

4. Every Red has 2 Black children (one or both of which could be nil.) Equivalently, Red nodes do not have Red children.

5. For any node x in T every descending path from x to a leaf has the same number of Black nodes.

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

Ex.




| node | black height |
|---|---|
| 8 | 3 |
| 5, 11, 3 | 2 |
| 1, 2, 4, 6, 7, 9, 10, 12 | 1 |
| all nil leaves | 0 |

absolute height

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

iff $x$ is a leaf (nil)

## Theorem

A RBT with $n$ internal (non-nil) nodes and height $h$ satisfies

$$h \le 2 \lg(n+1).$$

Recall: in any Binary Tree $T$:

$$\text{height}(T) \ge \lfloor \lg n \rfloor$$ where $n$ = # of internal nodes

$\therefore$ in any RBT $T$ with $n$ internal nodes:

$$\underbrace{\lfloor \lg n \rfloor}_{\Omega(\log n)} \le \text{height}(T) \le \underbrace{2 \lg(n+1)}_{O(\log n)}$$

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