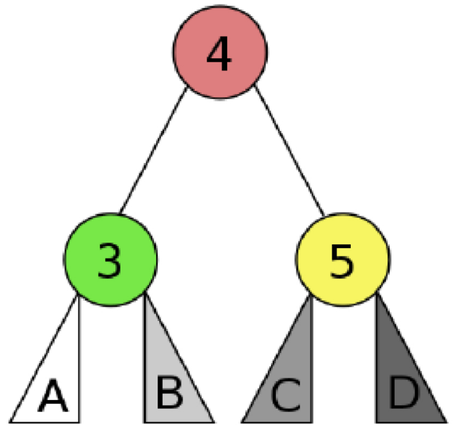
Announcements

- MP5 is due on Friday, 04/01 @ 11:59pm
- Exam 3: March 30 - Apr 1.
- Exam 3 solution party, Monday March 28 @9pm, location Siebel 1404
- Conflict exam: send email request to me, Thierry and Cinda.
- Exam 3 will cover: MP4, MP5.1, lab_trees, lab_dict, lab_huffman (due Apr 3)
  Iterators, Functors, Trees, Running time, Induction

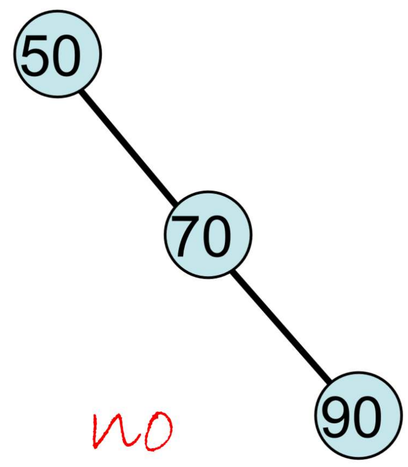
Balanced



http://www.qmatica.com/DataStructures/Trees/AVL/AVLTree.html

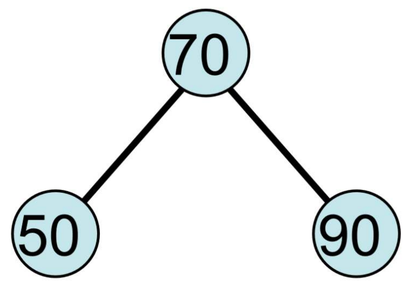
Reminder from last time:

b = 2



Height balanced? no

b = 0



yes

The “height balance” of a tree T is:

$$b = height(T_L) - height(T_R)$$

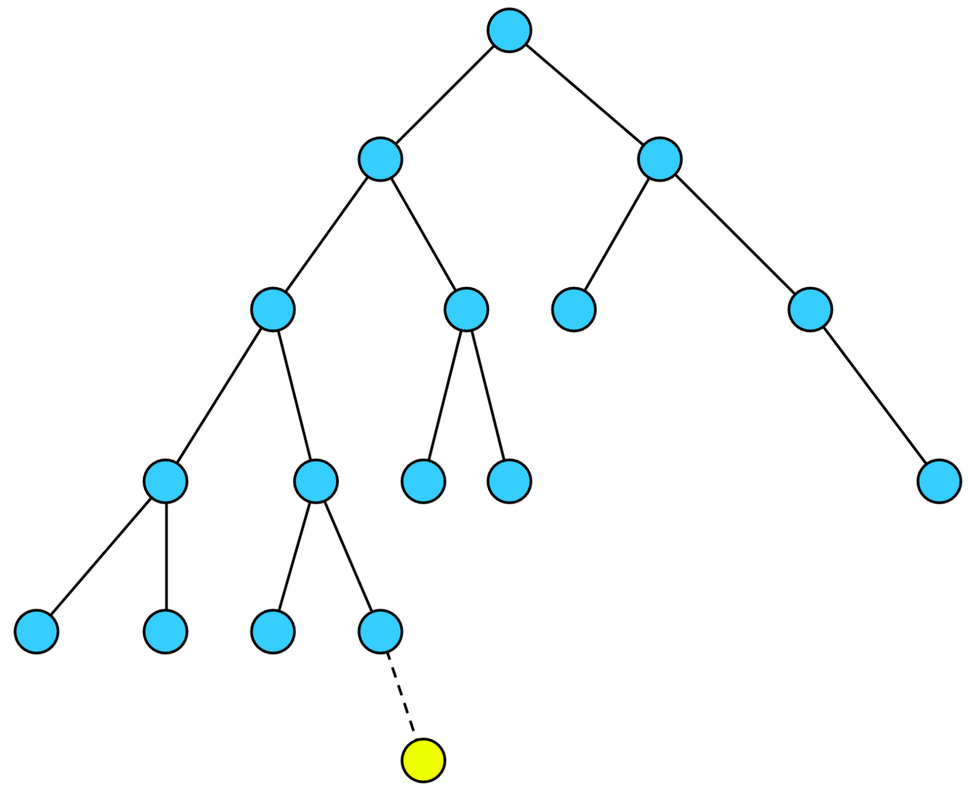
A tree T is “height balanced” if:

- T = {}
- T = {r, T_L, T_R},
  |b| <= 1, T_L and T_R are both height balanced

We call these trees "AVL trees"

AVL trees:

```
struct treeNode {
    T key;
    int height;
    treeNode * left;
    treeNode * right;
};
```



Insert:

insert at proper place

check for imbalance

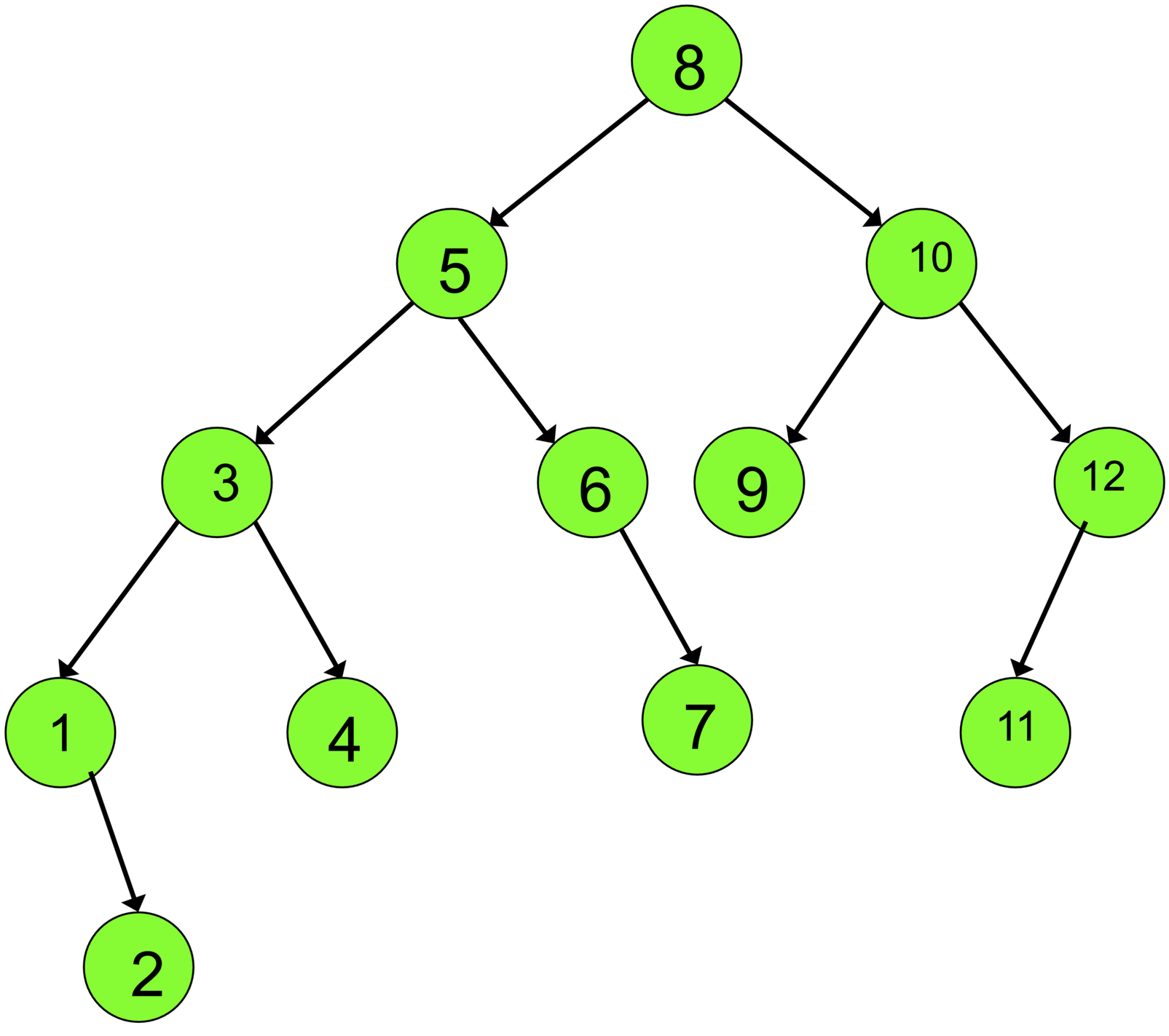
rotate if necessary

update height

AVL tree insertions:

```
template <class T>
void AVLTree<T>::insert(const T & x, treeNode<T> * & t ){
   if( t == NULL ) t = new treeNode<T>( x, 0, NULL, NULL);
   else if( x < t->key ){
      insert( x, t->left );
      int balance = height(t->right)-height(t->left);
      int leftBalance = height(t->left->right)-height(t->left->left);
      if( balance == -2 )
         if( leftBalance == -1 )
            rotate_____________( t );
         else
            rotate_____________( t );
   }
   else if( x > t->key ){
      insert( x, t->right );
      int balance = height(t->right)-height(t->left);
      int rightBalance = height(t->right->right)-height(t->right->left);
      if( balance == 2 )
         if( rightBalance == 1 )
            rotate_____________( t );
         else
            rotate_____________( t );
      }
   t->height=max(height(t->left ), height(t->right))+ 1;
}
```

AVL tree removal:

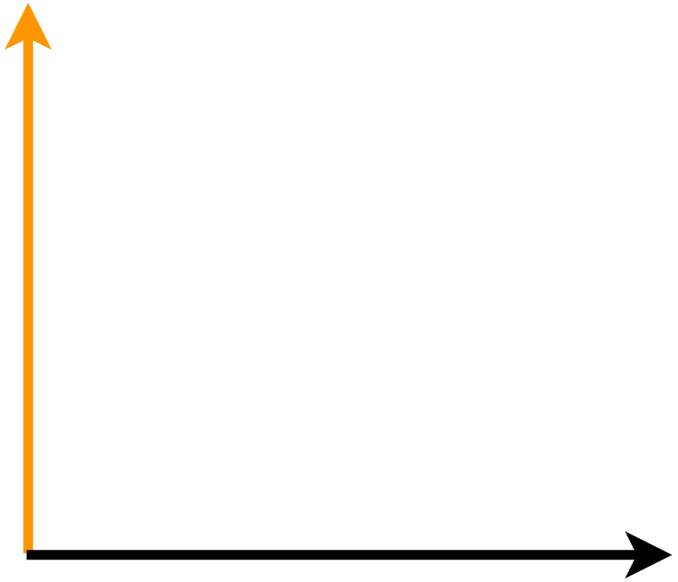


AVL tree analysis:

Since running times for Insert, Remove and Find are O(h), we'll argue that h = O(log n).

- Defn of big-O:

- Draw two pictures to help us in our reasoning:



- Putting an upper bound on the height for a tree of n nodes is the same as putting a lower bound on the number of nodes in a tree of height h.