Cmp 338

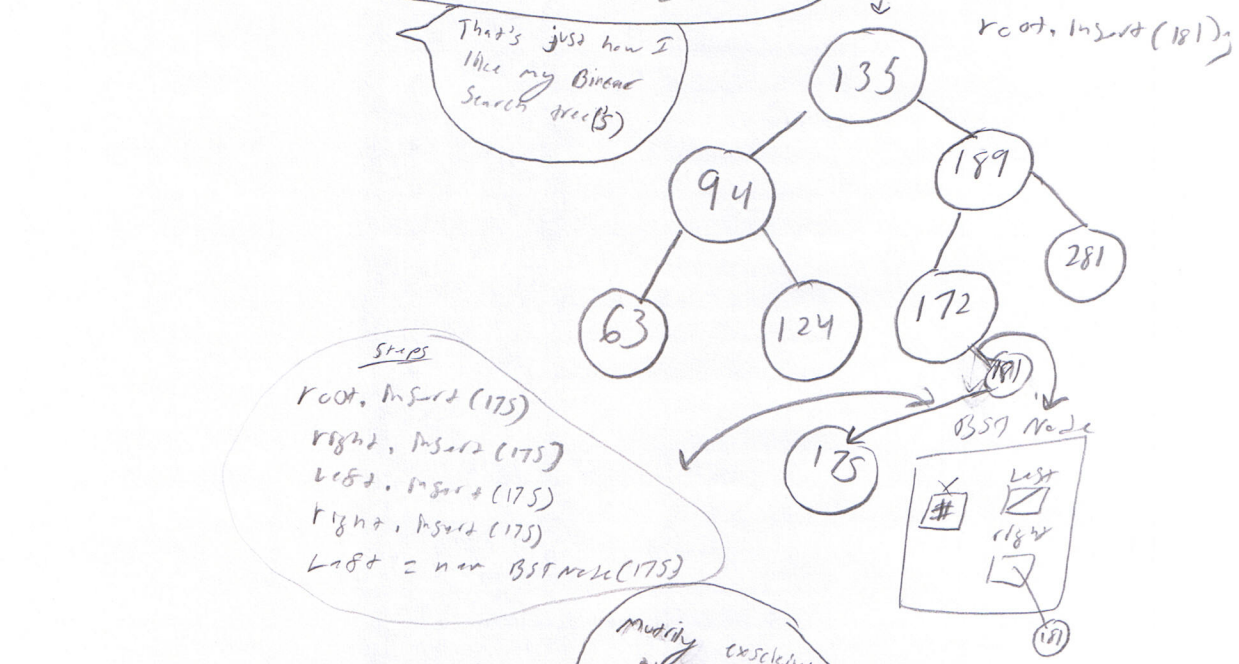
10·20·09

```
public void insert(int x){
                    →or→(x > this.x)
  if(this.x < x){
      if(Right != null){Right.insert(x);}
      else
  }
                  →or→(x < this.x){Right = new BSTNode(x);}
  if(this.x > x){
      if(Left != null){Left.insert(x);}
      else{Left = new BSTNode(x);}
  }
```

Smart count

note if x = a number all ready in the tree it will drop / do nothing / won't be added

That's just how I like my Binary Search tree(BST)



Steps
root.insert(175)
right.insert(175)
Left.insert(175)
right.insert(175)
Left = new BSTNode(175)

modify exclusive area
both if statement can't be true at the same time!!

If I do one I can't do the other!

CMP 338

10.20.09

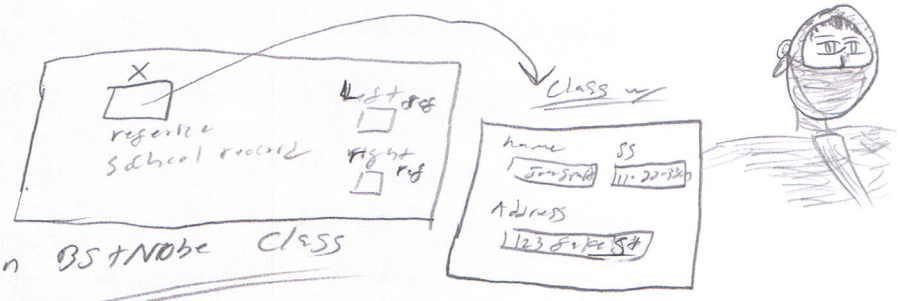
```
public class BinarySearchTree
{

    BSTNode root=null;

    public void Insert (int x)
    {
        if (root == null)  root = new BSTNode(x);
        else               root.insert(x);
    }
    public int nodeCount()
    {
        if (root == null)  return 0;
        else               return root.nodeCount();
    }
```

In order for the BSTNode class to work there has to exist a BSTNode

That's why Binary Search Tree is need



In BSTNode class

School record SearchC() → node

root

111-22-3344
John Smith
123 Sesame Street

```
public int Search (int x);
{
    if (x == this.x) return this.x;
    if ( x < this.x) return Left.Search(x);
    if ( x > this.x) return right.Search(x);
}
```

Code Complete (book)

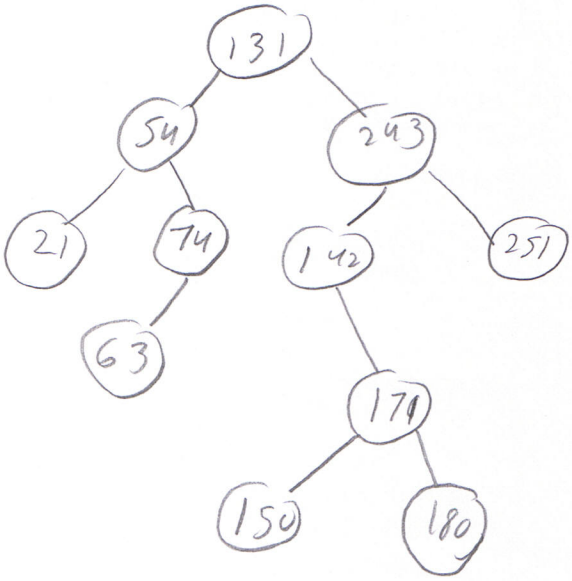
& Code Builders

Cmp 338

10/20/09

In Binary ~~Search Tree~~

```
public int search (int x)
{
    if (root == null)  // could return -1 or null (w/object)   return -1;
    else               return root.search(x);
}
```




```
root.Search (150)
right.Search (150)
Left.Search (150)
right.Search (150)
Left.Search (150) = 150
```

return 150 150 150 150