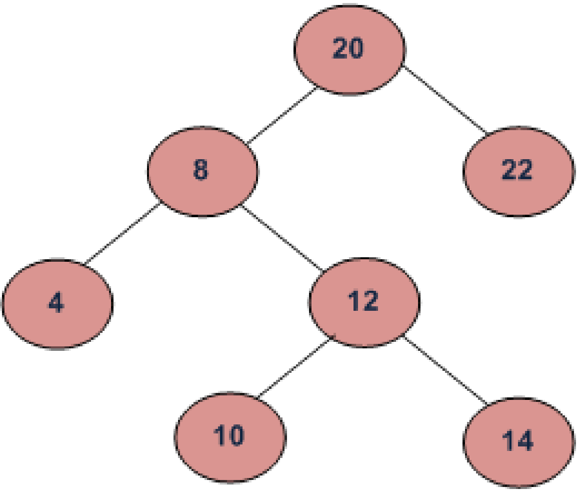


CSSE 230 Day 9

Binary Search Tree intro

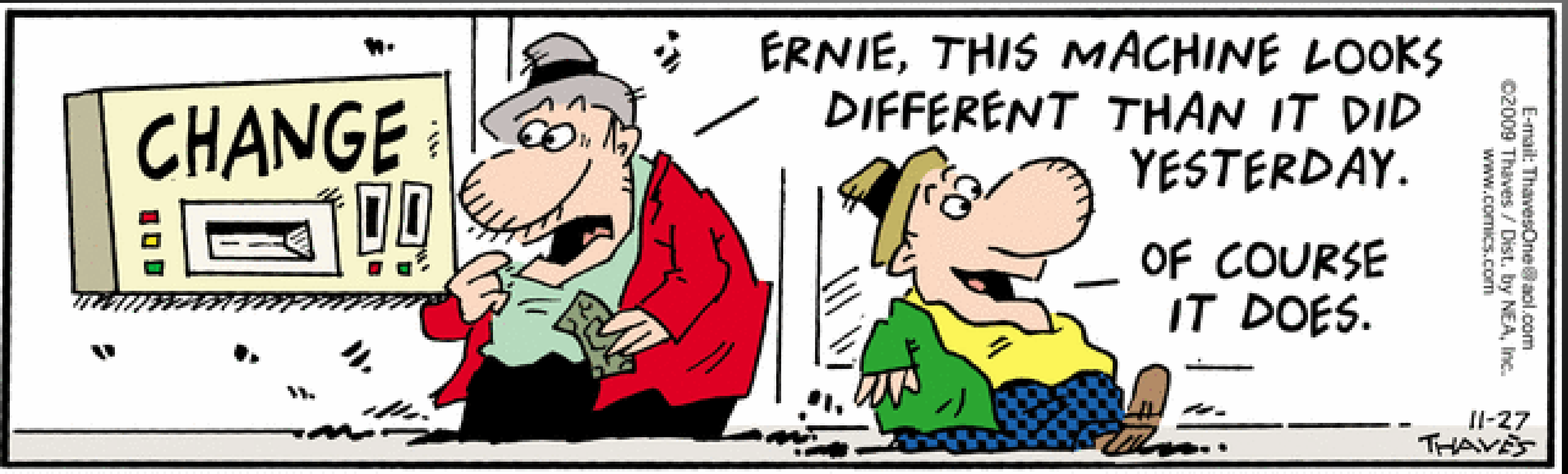
BST with order properties

Check out BSTNullNode project from SVN

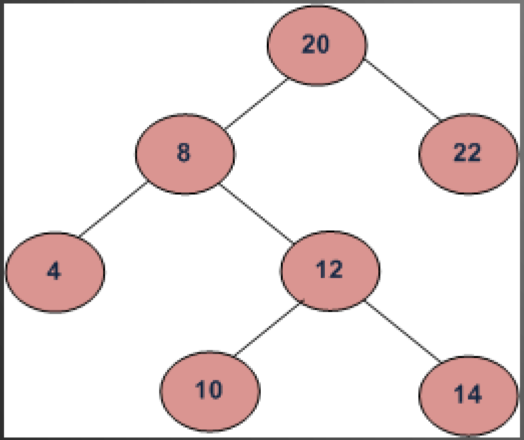
Announcements

- Partner Evaluation done?
- Displayable Partner Preference survey

Questions?


CHANGE
ERNIE, THIS MACHINE LOOKS DIFFERENT THAN IT DID YESTERDAY.
OF COURSE IT DOES.
E-mail: ThavesOne@aol.com
©2009 Thaves / Dist. by NEA, Inc.
www.comics.com
11-27
THAVES

Binary Search Trees



Binary Trees that store elements in increasing order

A Binary Search Tree (BST) allows easy and fast lookup of its items because it keeps them ordered

Q1

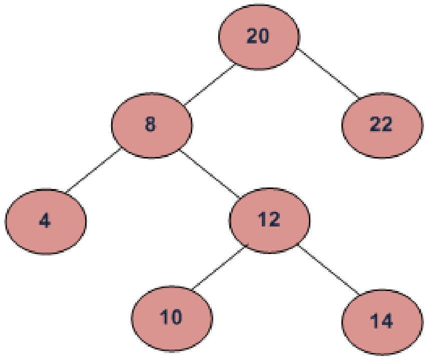
Draw a "birthday BST"

- A BST is a Binary Tree T with these properties:
    1. Elements are Comparable, and non-null
    2. No duplicate elements
    3. All elements in T’s left subtree are less than the root element
    4. All elements in T’s right subtree are greater than the root element
    5. Both subtrees are BSTs
- **Advantage:** Lookup of items is O(height(T))
- What does the inorder traversal of a BST yield?

BST insert, contains, and delete are different than in a regular binary tree

Q2-7

```
public class BinarySearchTree<T extends Comparable<T>> {

 private BinaryNode root;

 public BinarySearchTree() {
   this.root = null; // or NULL_NODE;
 }

 // insert obj. If already there, return false
 public boolean insert(T obj)

 // Does this tree contain obj?
 public boolean contains(T obj)

 // delete obj. If not there, return false
 public void delete(T obj)
```


20
8
22
4
12
10
14