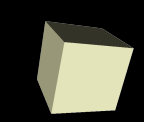
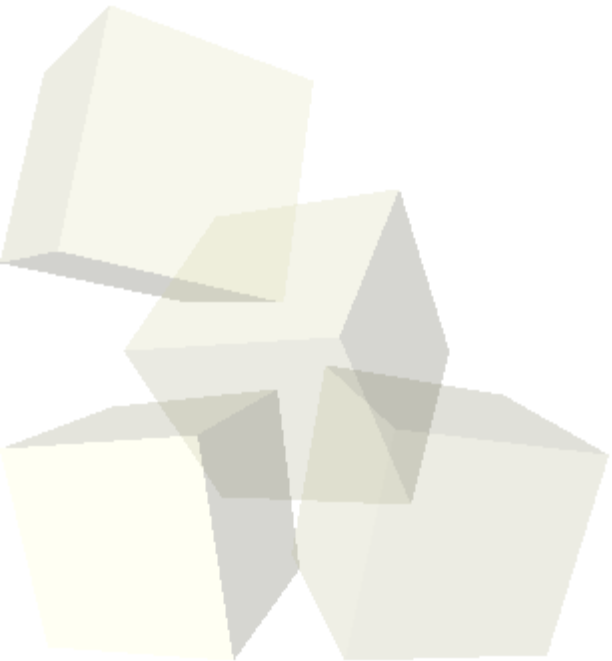


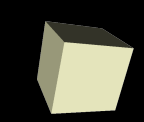
Binary Search Trees

4/9/2009



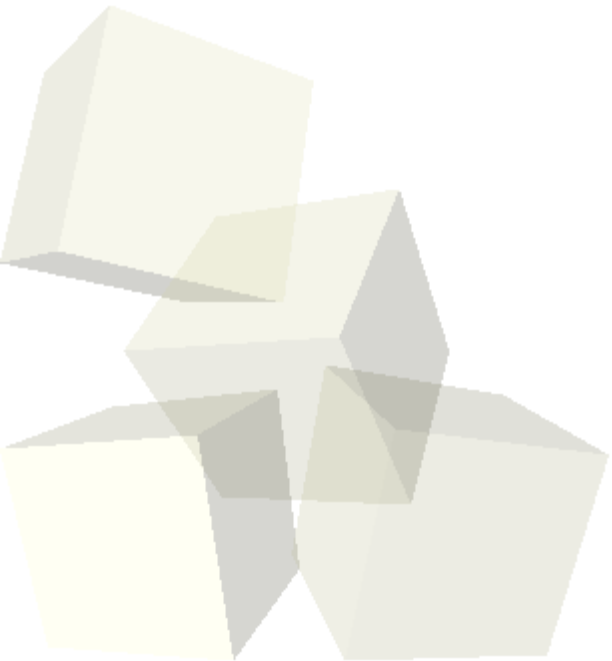
Opening Discussion

- Let's look at solutions to the interclass problem.
- Do you have any questions about the assignment?
- Do you have any questions about the reading?
- Minute Essay comments
  - How are trees used in advanced programming?
  - Why not traverse trees one level at a time?
  - Code for solving bin packing the i<binSize.length is used so we run through all the bins checking if the current item fits in any.
  - Uses of traversals. Goal is understanding.
  - Are recursive problems all representable as trees?



Code

- Let's begin by finishing up the code we started last time.
- We want our formula parser to parse to a tree structure so that we can evaluate the same formula many times quickly.



Binary Trees

- Sometimes we want to limit how many children a node has. One of the most commonly used trees in programming is the binary tree where no node has more than 2 children.
- The children are often called left and right.

In-order Traversal

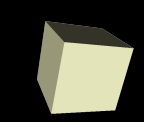
- For a binary tree there is an extra type of traversal called an in-order traversal where the node is visited between the recursion down left and right.
- Equations are great examples of trees. We typically write them out in the in-order. We could just as well write them out in post-order or pre-order.

Sorted Binary Trees

- One of the best uses of binary trees is the sorted binary tree. They make a more efficient implementation of the map ADT.
- In this type of tree, we store a key and data in every node and below any node we put lesser key values to the left and greater key values to the right.
- We find elements by going down the tree always going left or right. This gives us behavior like a binary search, but the tree is more flexible because adds and removes are quite efficient as well.

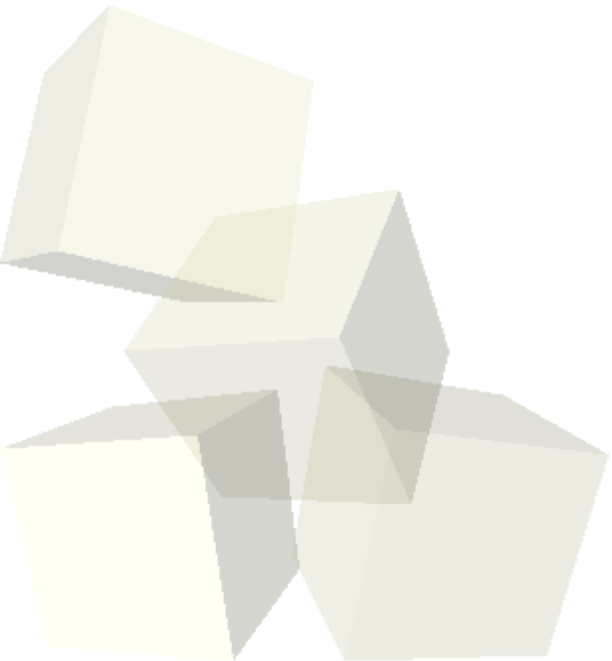
Adding and Removing

- The code for both adding and removing from a binary tree begins like a search that keeps track of previous (much like a singly linked list).
  - The add always goes to a leaf and adds the new element to the proper side.
  - The remove replaces the node we are removing with either the greatest node on the left or the smallest node on the right.



Coding

- I want us to code a BST based map together.



Minute Essay

- What can go wrong with the type of binary tree that we wrote today to make it perform poorly?
- Interclass Problem – Edit a Drawable so that it uses a formula for something instead of a regular double.