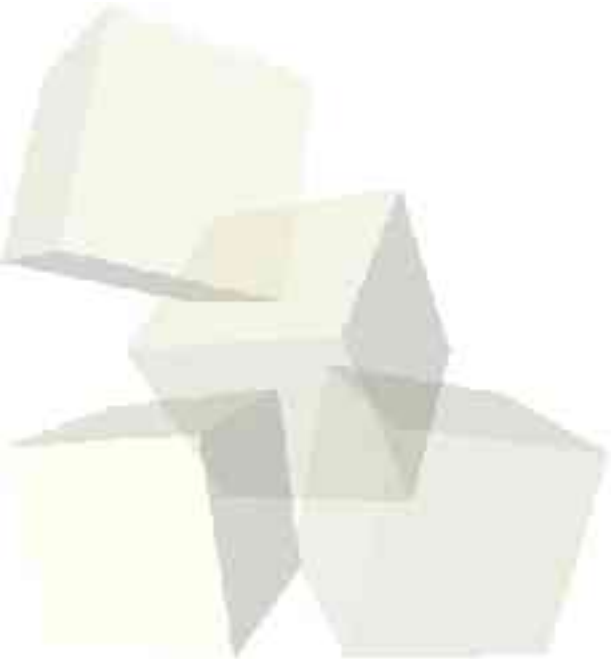
Binary Trees in ML

11-1-2004



Opening Discussion

- Do you have any questions about the quiz?
- What did we talk about last class?
- How many people did the binary search tree option for assignment #4? What were some of the key aspects of the implementation of that? How was it different from a binary search tree in Java or C? How do you think we could do it in ML?

Basics of BSTs

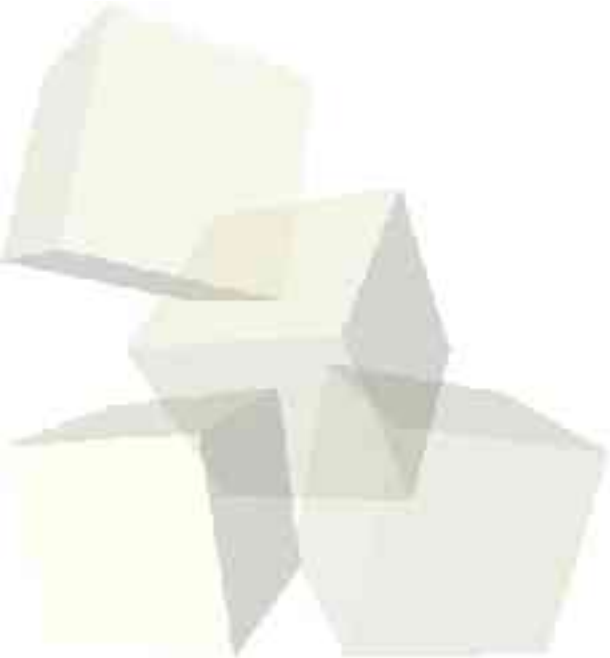
- A binary search tree is a very standard data structure in CS because it can store any sortable elements and allow inserts, removes, and searches to all happen in O (log N) time.
- The idea is that each node has a value associated and two “children”: left and right. All smaller values go left and all larger ones go right.

Keys and Datatypes

- Unlike your book I make a datatype that has two pieces of “data” in it. The first is what the tree is sorted on and searched by, a “key”. The second is the full data. The key can be some sub-element of the data like a name for a student. This only makes sense when looking things up.
- So our datatype has an Empty option as well and a BNode option with a 4-tuple of data in it.

Code for Binary Search Tree

- Now let's go write code to implement a binary search tree in ML. This should help you get a better idea of how to write and use datatypes in ML.



Minute Essay

- Assignment #7 is having you write some code that loads, edits, and saves XML documents. Are you familiar with XML? Have you ever done hand coding of HTML?
- Do you have any ideas for “interesting a fun” assignments?