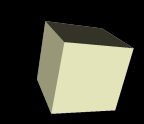
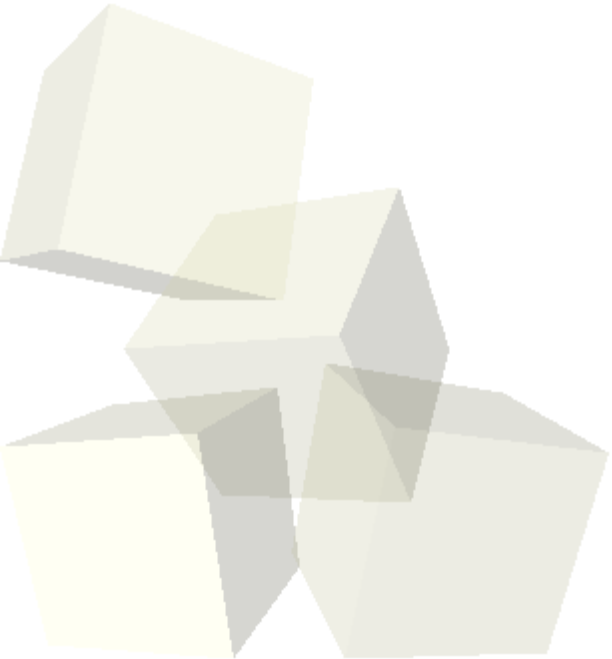


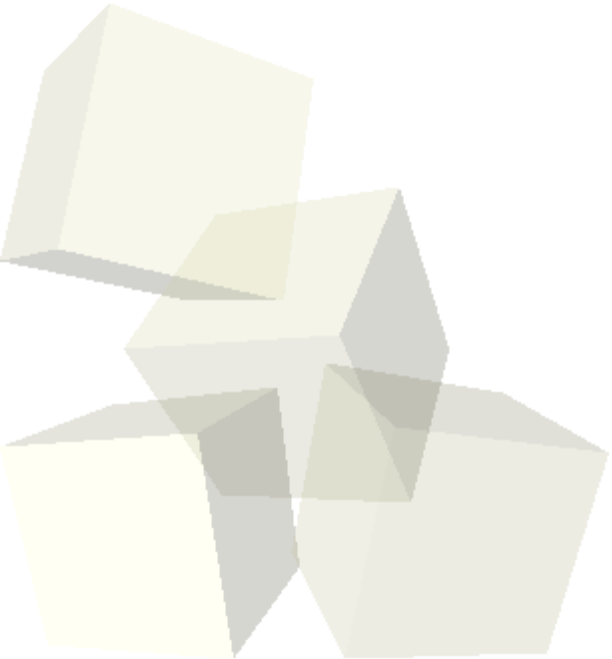
Linked Lists and Iterators

2/26/2008



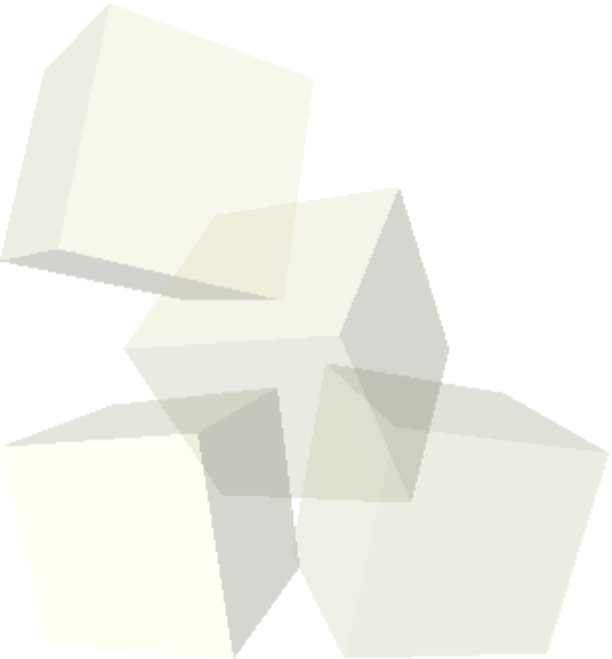
Opening Discussion

- Let's look at some solutions to the interclass problem.
- Do you have any questions about the reading?
- Do you have any questions about the assignment?
- Can you use linked lists like objects?



Implementing a Singly Linked List

- Last class we started working on our implementation of a singly linked list. Let's add a few more of the basic methods to that real quick.

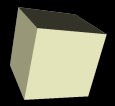


Sentinels

- Your book refers to an extra node placed at the beginning of a linked list as a dummy node. These are also called sentinels and your book understates how much they can improve your life. They also don't do them quite right.
- A sentinel is an extra node in the list the represents the “end” of the list and doesn't store data. The purpose of the sentinel is to remove special cases. The next of the sentinel is what we have called head.

Implementing a Doubly Linked List

- Now let's implement java.util.List with a doubly linked list with a sentinel. The list will also be circular.
- You should notice that this implementation never has to check for null because no references in the list should ever be null. This simplifies the code significantly. We also implicitly get a head and a tail with no extra work. If you don't have a sentinel you will write a lot of extra checks for nulls and even more to include a tail.

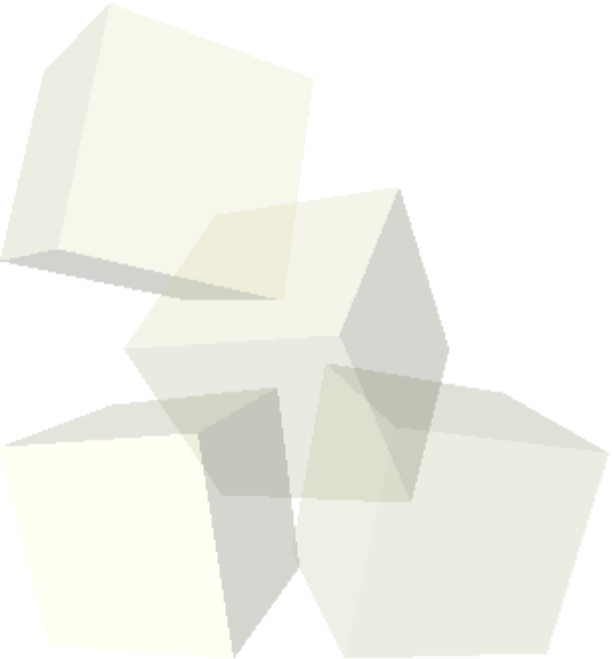


Iterators

- We've already discussed that direct access on linked lists is very inefficient. How then should we walk through a list with outside code? Remember that the outside code doesn't have access to the nodes so it can't use the style of loop we have been doing internally.
- The concept of an Iterator is something that abstracts the process of walking through all of the elements in a container. Iterators can not only be efficient, they also make code more flexible because they don't depend on the implementation details of the containers.

Iterating Lists

- An iterator basically needs to encapsulate the information and functionality we would put into a standard method of going through a container.
- With this in mind, what do we need to put in an iterator for an array based list?
- What would we put in an iterator for a linked list?



Minute Essay

- You have now seen linked lists done in both C and Java. What differences stand out?
- The design for assignment #3 is due today.
- Interclass problem – For this interclass problem I want you to pull together your stack based RPC and your list to make a simple RPC with memory. If the user enters a number or an operator it should do what the RPC normally does. If they enter “S” it should store the top of the stack on the list. If they enter “M” the next thing they enter should be an index and the memory at that index on the list should be pushed on the stack. If they enter “C” the list should be cleared.