Sorting, Searching, Stacks, and Queues

2-14-2003

Opening Discussion

- Do you have any questions about the quiz?
- What did we talk about last class?
- Do you have questions about the assignment?
- Private not only protects your data, but parts of code as well. It reduces dependencies so you are more free to change without breaking other code.

Cap Insensitive Sort with java.util.Arrays

- To do this, we need to use a Comparator. Remember, that Comparator is just an interface in java.util. We have to write a class that implements it for use with the sorts.
- Let's also "instrument" the code to count the number of comparisons and see how our sorts stack up against each other and theirs.

Abstract Data Types

▐ The main topics for today are Stacks and Queues. These are the simplest forms of what we call "Abstract Data Types". An ADT is basically an entity that stores data in some unknown form, and provides us with a standard interface for dealing with it.

▐ Good data structures in general are ADTs. They demonstrate the separation of interface from implementation.

The Stack and Queue ADTs

▐ The stack and queue are the simplest ADT because they each have only 2 methods in their interface (or 3-4 depending on the details of your definition).

▐ Stack
  ▐ Push: adds something new to the stack.
  ▐ Pop: removes the "newest" thing.

▐ Queue
  ▐ Enqueue: adds something to the queue
  ▐ Dequeue: removes the "oldest" thing.

LIFO vs. FIFO

▐ A stack is a last in, first out data structure.

▐ A queue is a first in, first out data structure.

▐ You will commonly here these referred to with the acronyms LIFO and FIFO.

▐ This really is the only difference between the two as for as a mental model of them is concerned.

Array Implementation of a Stack

- Stacks can be implemented in many ways. There is even a simple implementation with an array.
- By just keeping an integer index to the "top" of the stack we can easily write the methods.
- The third method often added is a peek method to look at the top of the stack without popping and an isEmpty method.

Array Implementation of a Queue

- As with a stack, a queue can be fairly easily implemented with an array, but here there are a few more details to worry about.
- Now we need two indexed for a front and back of the queue. Array based queues also need to be "circular" or they run out of space quite quickly.
- Again, peek and isEmpty are helpful.

Minute Essay

- Draw a picture of what the array for a stack would look like after these operations:
  - push(5)
  - push(7)
  - pop()
  - push(2)