Bookstore has books! If you prepaid, they sent you an email.

Monday is a holiday, so there won't be any lab on Monday. I'm going to ask the TA's to talk about AFS and RCS in lab, so you should try to attend on Tuesday or Weds.

On this week's homework, you may use my source code from the Card, Deck, and CardGame classes.

I'll post the next homework assignment over the weekend, assuming I'm healthy.

---

ADT homework not very good. I'm concerned that your programs are not going to be well-written based on what I have seen, and some of you may not even get them to work.

You should have worked with your partner!

Too many implementation details: arrays, for loops, private variables!!!

What's an ADT? What were some of the operations on an ADT List? Did we know anything about how the list was implemented?

Let's look at one or two methods that an ADT Hand for blackjack needs to provide.

**addCard**

  What should be passed to this operation? Card or Deck

**createHand**

**displayHand**

**getValue**

**setFirstCardFaceUp**

**setFirstCardFaceDown**

---

Reference types

What's a reference type?

How is it allocated in memory?
Can there be more than one reference to the same data?

What happens when you pass ref types as parameters?

reference assignment

reference equality

Linked Lists

Review of advantages/disadvantages of using arrays. Advantages: Fast access to elements

Possible solution: "resizable" arrays. Allocate a larger array when the current one gets full, copy all the elements, and then change the reference. Works, but 'expensive' operation for large arrays. java.util.Vector uses a similar technique.

We saw a conceptual linked list last time, showing how the data can be stored in non-contiguous memory.

Each of these 'memory' locations really has two parts: the item and a 'link' to the next item. Call these nodes.

We can implement each node as an object

```java
Node aNode = new Node( new Integer(6) );
Node anotherNode = new Node( Integer(9), aNode );
Node thirdNode = new Node( Integer(3), anotherNode );
```

Two things to note: there is a first and a last element. We use null as the reference value of the last element, so that we can tell when we’re at the end of the list.

Typically, we have a variable named head that contains a reference to the first element of the list.