Project #2B  
Stacks and Queues

You are to write a program that will handle the inventory for “cases of canned beans” for a supplier for a BiggerBox warehouse. For bookkeeping purposes, the management of the inventory will follow a LIFO structure. We will assume that the cans will not “expire” any time soon (so that actual sales can also follow the LIFO maintenance), that the cost of the beans will increase with time, and that the price charged will be based on the most recent (and therefore highest priced) shipment of beans received. Customers will be charged a 40% markup over the cost.

For bookkeeping, we will keep track of the amount charged to the individual store, and the actual cost of the cases that were sold. For example:

Shipments received:  
100 cases of cans @ $3.98  
50 cases of cans @ $4.03  
75 cases of cans @ $4.07

Store order for 200 cases.

The store will be charged 30% over the highest cost, $4.07 for the 200 cases =   
200 @ $5.29 = $1058.00 for the Customer receipt.

Actual cost: 75 @ 4.07 = 305.25  
 50 @ 4.03 = 201.50  
 75 @ 3.98 = 298.50  
Total cost = $805.25 for the bookkeeper’s records.

Shipments will be received (saved in the appropriate structure) and sold in the order requests are received. If there are not enough widgets in stock for a sale, the balance of the customer’s order will be placed on backorder (stored in an appropriate structure), and filled when the next shipment comes in, at the price they would have been charged, not the new price. The backorders should be filled before new sales are processed.

I will post a text file, transactions.txt with data like:  
R 100 3.98  
R 50 4.03  
R 75 4.07  
S 200  
On each sale, you should “display” the store’s receipt and the bookkeeper’s data.  
  
Each transaction should be processed as it is posted (read in)

You may use the Java data structures that are appropriate, the ones from the textbook, or the structures that you have already compiles, and your program should be organized! Your information on the shipments received should be treated as **objects**, not separate data (quantity, price).

The program is due on Tuesday,November 21 at 11:59 PM.