CSCI 230 – Program 1

Assigned: Thursday September 5th

Program Due: Friday September 13th by midnight CST

You have been hired by *Sammy's Warehouse Club* to create a program that will track a guest's shopping list. Each `Item` in the list has a name, a quantity and a unit price. This program lets us exercise the Ordered List ADT. You are to complete the implementation of the Ordered Fixed-Size Array implementation in which the ***add method inserts an item into its proper sorted slot*** in the array. This means that searches can take advantage of efficient binary search instead of sequential search. The print method simply prints the Items in the order in which they are stored since the add method will have inserted them into their correct sorted locations.

Items in our list will consist of the name of the item (as the key), the quantity of that item and the item's unit price. The key field is implemented using the Java String class.

```
Item()
Item(int number_of_items, String name_of_item, double price)
int getQuantity()
String getName()
double getUnitPrice()
String toString()
```

Define Java interface **`ItemOrderedList.java`** to include public method signatures of:

`public void clear()`: Resets the list to the empty state.

`public boolean add(Item product)`: Inserts the product into the list if its key is unique (case insensitive) and the list is not full. Returns true if successful; otherwise, false.

`public boolean delete(String keyValue)`: Deletes the Item with the given key (case insensitive). Returns true if an Item was found and deleted; else, false.

`public Item retrieve(String keyValue)`: Returns the Item having the specified key value (case insensitive); otherwise, return null if not found.

`public boolean isEmpty()`: Returns true if and only if the list is empty.

`public boolean isFull()`: Returns true if and only if the list is full.

`public void print()`: Prints all of the Items in the list in ascending order by the key field.

Write class **`FAItemOrderedList`** which implements **`ItemOrderedList`** interface. Your class will need to define a constructor **`FAItemOrderedList()`:** Default constructor creates an empty list. Initialize the capacity of your internal array implementation to hold at least 50 items.

Main driver program **`Prog1.java`** provides a menu as shown below. Please order and number the menu choices as shown below.

```
1. Add an item to the list
2. Delete an item from the list
3. Print each item in the list
4. Search for a user-specified item in the list
5. Determine whether the list is empty
6. Clear the list
0. Quit
```

Each menu command should display informative prompt messages as well as print a message indicating the result of processing that command. See the sample program run for example prompt and output messages. There is no data file for this program. All data will be entered via keyboard.

Program Comments

Each method of `ItemOrderedList.java and FAItemOrderedList.java` must have detailed header comments that specify input parameters, return value, and side effects that change the contents of the ADT. For example,

```
/*
 * add
 * Inserts given Item into list if the list is not full.
 * @param thing Given Item whose contents have been initialized.
 * @return true if Item is not null and list is not full; else, false.
 * post-condition: Item is inserted into OrderedList if successful.
 */
```

Code Re-Use

You may add any additional private helper methods inside class **`FAItemOrderedList`** if that helps you produce a more elegant design – avoids duplicating the same code in two or more methods.

Deliverables: Use OAKS to submit the following four Java source files `Item.java, ItemOrderedList.java, FAItemOrderedList.java,` and `Prog1.java`.

Grading Criteria

10	Item class
5	ItemOrderedList interface .java file
5	FAItemOrderedList Constructor
10	Add method
15	Retrieve method (must implement binary search)
15	Delete method
5	Test if list is empty method
5	Clear list method
5	Print contents of list including sorting
15	Prog1 Main with menu processing, prompt, and print feedback messages
10	Method header comments in ItemOrderedList.java and FAItemOrderedList.java

Sample Program Run

<prints menu – omitted for brevity>
Please enter your choice 0-6: 1

Please provide information about the item to add.
Please enter the name: Cereal
Please enter the quantity: 4
Please enter the unit price: 4.00

Cereal has been added to the cart.

<prints menu>
Please enter your choice 0-6: 1

Please provide information about the item to add.
Please enter the name: Bananas
Please enter the quantity: 3
Please enter the unit price: 0.80

Bananas has been added to the cart.

<prints menu>
Please enter your choice 0-6: 5

Your cart contains a total of 7 items.

<prints menu>
Please enter your choice 0-6: 2
Enter name of item to delete: Cereal
Cereal was deleted from your cart.
<prints menu>
Please enter your choice 0-6: 3

The shopping cart contains:
Cereal
quantity: 4
unit price: 4.00