# Programming Assignment #2

Manage a Small Electronics Store Inventory

CS 2308.255 and 256, Spring 2013
Instructor: Jill Seaman

**Due:**
**section 255:** in class **Tuesday, 2/12/2013** (upload electronic copy by 4:00pm)
**section 256:** in class **Monday, 2/11/2013** (upload electronic copy by 1:00pm)

---

**Problem:**

Write a C++ program that will allow a user to manage the inventory of a store that resells used electronics.

The inventory for the small electronics store will contain the following information for each item in the inventory:

| | |
|---|---|
| SKU | (stock-keeping unit, an integer, could be ANY integer) |
| quantity | (how many of this item in stock) |
| price | (in dollars and cents) |
| make+model | (i.e. "Apple iPhone 3GS 8GB", may contain spaces in it) |

Note: You may assume the store has less than 10,000 skus.

The program should offer the user a menu with the following options:

1. Add a new item to the inventory (prompt user for input values).
2. Remove an item from the inventory (by sku).
3. Display the information for an item (given the sku).
4. Display the inventory sorted by sku.
5. Quit

The program should perform the selected operation and then re-display the menu.

Do not change the menu numbers associated with the operations.

Note for options 1 and 2, you are not changing the quantity of an item. You are adding (or removing) the information about an item from the inventory (you are adding or removing something from an array).

For option 3, if the item is not found, display an appropriate message.

For option 4, display the information for each item on a separate line. The values do NOT need to line up in columns.

**NOTES:**

- This program DOES need to be done in a Linux or Unix environment.
- The program must be modular, with significant work done by functions. Each function should perform a single, well-defined task.
- Use an array of structures to store the inventory in the program.
- You may use the code from the book in chapter 5 (Program 5-8) as a template for the menu portion of the program. I have put this code on the TRACS site under resources.
- For the search and the sort, you may use any of the search and sort algorithms that we studied in class. I have put the code from the lectures on the TRACS site under resources
- I recommend implementing the features in this order: Add an item, Display the inventory (but unsorted), Display the information for ONE item (by sku), Display the inventory (sorted), then Remove an item.

**Logistics:**

Your program must be free of compiler errors.

Do not use any features of C++ that we have not yet covered in class (use features from Chapters 1-8 and 10-11 only.

Name your file **assign2_xxxxx.cpp** where xxxxx is your TX State NetID (your txstate.edu email id). The file name should look something like this: assign2_js236.cpp

There are two steps to the turn-in process:

1. Submit an electronic copy using the Assignments tool in TRACS no later than one hour before class the day the assignment is due (see top of page 1).
2. Submit a printout of the file at the beginning of class, the day the assignment is due. Please print your name on the front page, staple if there is more than one page.

   If you are unable to turn a printout in during class, you have until 5pm on the day the assignment is due to turn it in to the computer science department office (Nueces 247). They will stamp it and put it in my mailbox. DO NOT slide it under my office door.