Makenzie Brian

October 23, 2015

CS 162

Assignment 2- Design and Testing with Classes

**Understanding the Problem:**

Make two classes: one for items, one for lists. Make the item class have a dynamic array of items. The item class will need a name, unit type, unit price, and number to buy (later include extended price). Item class needs accessors and mutators. The list class will need functions to add items, remove items, and print the list (all info for each item and the total list price). Overload the == operator to perform the test. If an item is being added, check to see if it is already in the list. If it is, give the option to change the quantity. Says to assume the user inputs data correctly so no input checking.

**Devising a Plan:**

Main will need a menu of options (from the list class functions and an exit option). Make an add function in the item class to set values of info when new item is added (easier to access the variables here). The list class will also need a search function to help when adding a function (if it already exists, change number instead of adding another item)(set name here if not already in array). Use bool values in loops to then use outside loop for determining if strings are the same (keeps things that don’t need to be repeating from repeating).

**Looking Back:**

I had issues with my values for k in my loop in my remove item function for my list class but it turns out I was going outside the scope for the array. Most of my other issues were syntax.

**Test Plan:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Case | Input | Expected Output | Actual Output | Reason |
| 1 | (1), a, unit, 1.1, 1, (3) | Prints a, unit, 1.1, 1 | Prints a, unit, 1.1, 1 | normal |
| 2 | (2), a, (3) | Removes a, prints nothing | Removes a, prints nothing | normal |
| 3 | (1), b, cluster, b | Seg faults | Seg faults | No input checking |
| 4 | (a) | Seg faults | Seg faults | No input checking |
| 5 | (4) | Exits program | Exits program | normal |