**Ecommerce System**

**Programming Project 4**

**Pseudocode Due: May 3rd 11:59 p.m.**

**Project Due: April 19th 11:59 p.m.**

Extend your first 3 projects to include classes. You must include all the feature of previous projects and upgrade your project features using the details below.

**Classes Required:** The members and functions to be included in each class are listed below. Other functions may be added to the class. Please remember that class functions should be generic enough to be used in any situation. Functions that solve a specific problem for this project should be defined in main rather than the class.

* Linked List
  + Head (Inventory Pointer)
  + Default Constructor
  + Overloaded Constructor (pass in node which becomes new head)
  + Overloaded += operator
    - Adds node to end of the list.
* Inventory
  + Item Number (int)
  + Name (string)
  + Cost (double)
  + Qty (int)
  + Inventory Pointer (to point to next inventory item)
  + Default constructor
  + Overloaded constructor
  + Methods
    - Get the Item Number
    - Get the Name
    - Get the Cost
    - Get the Qty
    - Set the Qty
    - Set the Cost
    - Add Item
    - Delete Item
* Customer
  + First Name (string)
  + Last Name (string)
  + Address (string)
  + City (string)
  + State (string)
  + Zip code (int)
  + Invoice Filename (string)
  + Default Constructor
  + Overloaded Constructor
  + Accessor and mutators to update private customer data.
  + Shopping Cart (utilize the Linked List to create shopping cart).
  + Display shopping cart.
  + Add item to shopping cart
  + Delete Item from shopping cart.
  + Update Qty of item in shopping cart
* Admin
  + User Name (String)
  + Password (string)
  + Update Inventory Methods
    - Add Product
    - Delete Product
    - Update Qty
    - Update Cost
  + Validate username and password

When you program starts, create a version of admin which contains a user name and password. Ask if you are a customer or admin. If they select Admin, then they need to validate they can use admin tools by entering a username and password. Validate the username and password if it is correct then allow them to access the admin interface.

If they select customer, then create a version of the customer class and allow them to use the shopper interface.

Interface, create a nice interface to display menus of options users can pick:

***Example:***

Main Interface: Choose an Option:

1 – Admin or A – Admin

2 – Customer or C – Customer

3 – Exit or E – Exit

Custom Invoice Files: Use the last name of the customer in the file name “Lastname\_Invoice.txt”. Every time you have a customer, they get their own invoice file.

Notes:

* This project will utilize multiple classes (see above).
* Each class must have a header file and a source file.
* Remember the user can add as many items to their cart as there are in the inventory list.
* When updating the csv file should be the same name and same format as when you first loaded the file.
* **Remember to delete any allocated memory at the correct times of your program! Failure to do this will cause a reduction in your grade.**

**Pseudocode:**

* Turn in a word document or pdf containing your pseudocode for this project.
  + Your pseudocode should be in the correct format.
  + Add project part 4 logic to your updated pseudocode.
* Main.cpp
  + List functions you plan to create that are required (Remember these should only be functions that do not fit in the classes).
    - Determine the parameters
    - Determine the return type
    - Detail the step-by-step logic that the function will perform
  + Detail the step-by-step logic of the main function
* Overloaded += Operator.
* Use UML diagram to describe the Linked List Class.
* Use UML diagram to describe the Inventory Class
* Use UML diagram to describe the Customer Class
* Use UML diagram to describe the Admin Class

**Project Submission and Grading:**

1. Submit your assignment to eLearning.
2. Zip all of the source files and .h files into a single zipped file.
   * Turn in one file named “netid.zip” containing your program.
   * Make sure the zipped file has a .zip extension (not .tar, .rar, .7z, etc. You will be deducted points for incorrect file format).
   * Please make sure it compiles and runs correctly.
   * Use detailed comments inside your program explaining what each part of your program does and descriptions of all variables.
   * To get full credit you must implement more then one function. So, do not turn in a program with only a main function.

1. Turn in a file named “LastNameInvoice.txt” which contains a sample invoice you created from running the program. It must contain a minimum of 10 items (when the project is graded we may enter more or less items).
2. Turn in a file named “ProductData.csv” which is different from the version I gave with the project (new products added, qtys updated, old products removed).

**This is an individual project so please remember not to share your code with any of your class mates!**