CS 211 PA #1: Inventory Management

For this assignment, you will develop an inventory management system. Your application will maintain the text file "inventory.csv" and will allow users to add, update, and delete items in the inventory.

This assignment is an attempt to leverage concepts introduced in CS 112 with a more intricate system design that employs elementary data structures (i.e. vectors).

CSV File Format

CSV (comma separated values) is a standardized clear-text format frequently used to represent spreadsheets. All popular spreadsheet applications (e.g. Excel) are capable of opening and manipulating CSV files. The CSV format is pretty straightforward. Each line represents a row in the spreadsheet. Each cell in a row is separated by a comma. Our inventory CSV uses the following format:

NAME, QUANTITY, PRICE

The first row of the CSV contains column headers (Name, Quantity, and Price). Each subsequent row contains a single item in our inventory.

Program Flow

Your program should operate as follows:

1. Load the contents of inventory.csv into one or more variables. Vector usage is strongly suggested.
2. Provide the following menu options to the user:
    a. 1. Add item
        i. Prompt the user for an item name, quantity, and price. Add this item to the inventory. Ignore items with duplicate names.
    b. 2. Update item
        i. Prompt the user for an existing item name, quantity, and price. Update accordingly. Ignore items that do not exist in the inventory.
    c. 3. Remove item
        i. Prompt the user for an item name. Remove that item from the database.
    d. 4. Quit program
3. Upon receiving a request to exit, your program must then write the updated inventory back to inventory.csv

Header Comment, and Formatting

1. Be sure to modify the file header comment at the top of your program to indicate your name, student ID, completion time, and the names of any individuals that you collaborated with on the assignment.
2. Remember to follow the basic coding style guide. A basic list of rules is included with this document.

Reflection Essay

In addition to the programming tasks listed above, your submission must include an essay that reflects on your experiences with this homework. This essay must be at least 350 words long. Note that the focus of this paper should be on your reflection, **not** on structure (e.g. introductory paragraph, conclusion, etc.). The essay is graded on content (i.e. it shows deep though) rather than syntax (e.g. spelling) and structure. Below are some prompts that can be used to get you thinking. Feel free to use these or to make up your own.

- Describe a particular struggle that you overcame when working on this programming assignment.
- Conversely, describe an issue with your assignment that you were unable to resolve.
- Provide advice to a future student on how he or she might succeed on this assignment.
- Describe the most fun aspect of the assignment.
- Describe the most challenging aspect of the assignment.
- Describe the most difficult aspect of the assignment to understand.
- Provide any suggestions for improving the assignment in the future.

Deliverables

You must upload your program and reflection as a ZIP file through Canvas no later than midnight on Sunday, September 9 2018.

Grading Criteria

Your assignment will be judged by the following criteria:

Reflection essay (5pts)
- Your reflection meets the minimum requirements as specified earlier in this document.

Test Cases (10pts / ea)
- In order to receive full credit, your assignment must pass a series of test cases. All test cases are provided in advance. Thus, it is possible for you to know with great certainty what your grade will be before you submit your assignment.
- Each test case is worth 10 points. There are 6 test cases in total. The answers for the first 3 test cases are provided.
- To run a test case in the Windows Command Prompt, MacOS, or Linux, you can type (assuming your program is called pa1.exe): "pa1.exe < test1_add.txt"
- To run a test case in Windows Powershell, you can type: " GET-CONTENT test1_add.txt | ./pa1.exe"

Grade Distribution

Your final grade for the assignment will be determined based on the number of points earned.

Score (%)	Points Required
100	85
90	80
80	70
70	55
60	40
50	25
40	20
25	15