3460:209 Assignment 6-B

# **Assignment 6-B: Farmer’s Market I**

**Overview**

The purpose of this assignment is to make sure that you know how to write a program that uses files and file processing. The program also contains functions and does input, output, flow of control and/or calculations.

**PROGRAM SPECIFICATION**

For the assignment, we will write a program that reads a file of information pertaining to a Farmer’s Market. The file contains the names of various farms participating in a Farmer’s Market. The program should open and connect to a file object for input, read the data, perform some calculations, create a report, and then rewrite the information to a new output file.

The format of the file is as follows:

|  |  |  |
| --- | --- | --- |
| Field or data item in the file | Data type | Notes |
| Farm name | String of characters | This could end with a comma |
| Count of item | Integer | Number of items |
| Item | String | Name of the item |
| Item price | Float or double | Unit cost of each item |

Download the file (called ASSGN6-A.txt) to examine the contents. You should see that the name of the farm is delimited with a comma, and everything else is delimited with a space. When reading the information into your program, the comma after farm name and the space between each remaining field is consistent and guaranteed. Item is also guaranteed to be one word (so probably don’t need a getline). Also, we will not need to validate the information in this program. Valid data is also guaranteed! Your program will read the data items and output them to a report. The count of items should be multiplied by the item price, and that should be shown as well.

The format should appear on the screen as follows:

==============================================================

= FARMER’S MARKET INVENTORY =

==============================================================

Collins Farm 43900 tomatoes @ .67 each totaling $29413.00

…

Rutherford Farms, Inc. 809 apples @ .90 each totaling $728.10

…

Grand total: 999999 items totaling $999999.99

*Hint:* Use setw() in conjunction with setfill(‘=’) for your headings. Use setw() to align your output for the farm name. The maximum farm name is 25 characters in length. The remainder of the line will be formatted as seen. Also, try this ... << (farm + ":") << ... in your output statement to get the colon attached to the farm name. This ‘concatenates’ the string farm with the literal to make them one. We’ll cover this in detail later this semester.

Read each item and output them to the console as you process the file. Your report will show every item in the file.

As you process this file, you will ALSO be creating an output file. Your program should open an output file object using the name of your choice for the text file, and name of your choice for the name in your program. Rewrite each line of the input file to the output file, adding a variable with the amount that you calculated from count of items multiplied by the item price. Make sure that you put a space before the new total. As an example, the first item should look like this in your file:

**Collins Farm 43900 tomatoes .67 29413.00**

* **or –**

**Collins Farm 43900 tomatoes 0.67 29413.00**

Do both the report and new file creation in the same loop. Otherwise, your program will be very inefficient. Make sure you break your code into functions. Open your files in main and pass them by reference.

Make sure that your programs follow good documentation standards and follow the requirements for assignments. Reference the rubric standards on Brightspace. Note functions and data validation are now required. Do not use using namespace std;.

Submission Instructions – for programming solutions

On Brightspace, go to the matching Assignments for the **ASSGN@-#**, where @ is the chapter and # is the number or character of the problem assigned (eg., 5-11 for chapter 5, problem 11), and submit the program (cpp) and any (hpp) files.

*Last updated 5.22.2016 by Will Crissey.*

*Be aware that programming falls under all of the rules of plagiarism. Be careful when using any coding found in the outside world that is not your own. Any evidence of plagiarism is subject to sanctions like forfeits, suspension, and even ejection, as determined by the Department of Student Conduct and Community Standards.*

**COPYRIGHT © 2017 Will Crissey, Jr., Assistant Professor, All rights reserved.**

This document is intended exclusively for use by The University of Akron's Department of Computer Science. Do not distribute document copies to the public in any way.

***THIS PROGRAM'S CONTENT IS PROTECTED***

The document and associated content are either the property of, or used with permission by, The University of Akron, Department of Computer Science and may be protected by copyright and other restrictions. Copyrights in the site content may also be owned by individuals and entities other than, and in addition to, The University of Akron, Department of Computer Science. The use of this document by you, or anyone else authorized by you, is prohibited, except for purposes of completing the homework/project. Any unauthorized use of the document may violate copyright laws, trademark laws, the laws of privacy and publicity, and communications regulations and statutes.