Programming Assignment 6: Soft Drink Vending Machine Simulation

COP 3014 - Fall Term 2009
Point Value: 100 points

Project Due Date: Tuesday December 1, 2009 at 11:59 PM

NO programs will be accepted after this due time !!!
i.e. NO LATE PROGRAMS AND NO EXTENSIONS !!!

Learning Objectives

- To write a program which requires the creation and manipulation of an *array of structs*
- To utilize the *typedef* construct to create and work with the array of structs
- To utilize C++ strings to work with textual data (the C++ string class)
- To practice more advanced file input
- To make use of a standard sorting algorithm, the *selection sort*
- To simulate a common real-world process which is typically controlled by software

Problem Statement

Write a program to simulate a soft drink vending machine. The program will work with drink names, their prices, and the current number available in the machine.

First, you will read the initial machine state information from a data file named *machine.txt*. This file will contain the drink names, the price per can for each drink, and the number of cans available in the machine initially for each drink. If the initial data was given by that in this table,

Drink Name	**Price Per Can**	**Number in Machine Initially**
Cola	0.50	5
Ruby Red Blast	0.80	10
Lemon Lime	0.75	8
Grape Soda	0.85	20
Citrus Cooler	0.85	0
Sweet Tea	1.00	11

the text file would contain:

```
Cola<eoln>
0.50   5<eoln>
Ruby Red Blast<eoln>
0.80   10<eoln>
Lemon Lime<eoln>
0.75   8<eoln>
Grape Soda<eoln>
0.85   20<eoln>
Citrus Cooler<eoln>
0.85   0<eoln>
Sweet Tea<eoln>
1.00   11<eoln>
<eof>
```

Notice that the machine can hold six drinks with a maximum of twenty cans per drink.

In this project, your program must read in and store the machine data in an ***array of structs***, where each struct contains

- drink name (C++ string class object, may contain blanks)
- price (double)
- number of cans available (int)

Each time the program runs, it will enter a loop that performs the following steps: a list of drinks and their prices is displayed on the screen. The user is then allowed to either quit the program or pick a drink. If the user selects a drink, the program will notify the user that $1.00 must be entered for payment. The program will then display the amount of change that would be returned and subtract one from the number of drinks left in the machine for that drink. If the user selects a drink that is sold out, a message to that effect will be displayed. The loop then repeats. When the user chooses to quit the program, the program must display the total amount of profit the machine earned during the current run. The profit on each can is calculated as $0.25 per can, for all drinks.

After the user has quit and your program has printed out the total profit, it must then sort the array of structs containing the drinks into descending numeric order according to the number of cans of each drink which are available. Then, print out a summary table of the machine's final state, which includes the drink names, prices, and cans available in the machine at that point.

Input

Input is both interactive and from a data file in this project. You may assume the data file is not empty and is completely correct.

You may hardcode the name of the input file directly into your program. You must implement a check to see if the file specified does open, and if it does not, terminate the run using either return in main or exit in another function, as discussed in lectures. If this occurs be sure to print out an appropriate error message for the user.

User input consists of an integer representing a choice of drink (1 through 6), or the command to quit (0). You may not assume that the user enters an integer when prompted; hence you must check for the fail state and recover as needed.

The program must also perform bad data checking on the value of this user input item, and re-prompt the user until a valid menu choice is entered.

Output

For this project, you will be required to design the appropriate output yourself. Use your imagination. Follow the style guidelines for output we have been discussing and demonstrating all semester. Here are some basic ideas to keep in mind:

- All input data must be echoprinted
- All important values calculated must be printed
- You are simulating the interaction between a person using a drink machine and that machine; if you were buying drinks, what would you need to see displayed to understand what was happening at each step? What would you *like* to see to make the machine user-friendly?

Be sure to check your output for correctness using the testing and debugging techniques we discussed in class.

Use Of Functions

Part of your grade on this programming project will be determined by how well you utilize functions and parameters appropriately. Start by working on a good design, structure chart, etc. Your program should contain at least 7 functions other than main, and represent a modular, cohesive design following class style guidelines.

Use Of Data Structures and Data Types

You must use an *array of structs* as the major data structure on this project with each struct as specified earlier in this write-up. You must use the C++ string class to store and manipulate drink names.

You should only have one array of structs declared in this program. You do not need any other arrays to complete the task as specified (with the exception of the C++ strings discussed above). *You may not use* any two-dimensional arrays or user-defined classes on this project. You also *may not use* the C++ Standard Template Library classes or algorithms (with the exception of the C++ string class as described here).

Regarding FSU Computer Competency Requirement

This assignment is assessed according to the FSU Computer Competency Requirement. As such students must demonstrate (1) competent use of a discipline-specific software package: Microsoft Visual Studio C++ and (2) the ability to perform simple transactions using the web/Internet: obtaining all of the assignment materials from the class web site and downloading the required file(s), and turning in the completed work correctly to the class web site.

What File To Turn In and How to Turn In Your Work using Blackboard

Turn in your C++ program source file, which must be named ***machine.cpp***

Be 100% certain you have ***thoroughly*** read the handout entitled ***"Submitting Your Program Assignments Electronically Using Blackboard"*** for complete instructions on how to submit and how to **verify your submission** after you submit it. This will help you to avoid losing large numbers of project points due to improper submission issues. This handout is available on the course web site under "Handouts.""

Last Update: October 28, 2009 10:20 AM A. Ford Tyson
