Programming Project Report

Name: Jan Bobda
Date: 11/02/2018

Academic Integrity Statement: I pledge that I have neither given nor received unauthorized help on this programming assignment.

Problem Statement:

The goal of this programming assignment was to gain experience using one dimensional arrays in C++. I did this by simulating the user interface of an online food purchasing system. The program opens up with a menu, prompts the user for their name, and asks them what snack they want. The user responds with their name, the snack the want and how much of that snack they want to purchase. The program will continue to add snacks to the user's order until they decide to pay, at which point the program will print out the user's total cost. The program was required to check if the snack that the user entered was valid, and was still in stock. The program also had to check if the user entered a negative quantity or entered a quantity that was greater than the amount for the snack they had chosen.

Design:

The first design decision of this program was to decide what snacks to use and at what price to sell them. I also ended up storing all of my data in three separate one dimensional arrays, all the same size to keep track of the snack name, snack price, and snack quantity. I split the program up into four functions and the main function. One function was for printing the menu to the screen, another was for printing the inventory, one was for choosing a snack, and the last one was to find a matching item in a given array. The positives of this approach where that my code was easily reusable when splitting it into functions and cleaned up the general program. The cons to this approach was that I had to constantly pass the arrays through function headers because global arrays were prohibited for this assignment.

Implementation:

For this program, I started with a blank piece of paper so I had to incrementally build towards a working program. I started with the blank main function, added a cout statement and made sure that it properly compiled. Then I added the arrays of snacks, prices, and quantities and created a menu that printed them all out and submitted that for the midterm. From there I went ahead and began fulfilling all of the requirements outlined in the assignment one by one. Once I had completed those, I went back and added error checking where it was required. This process took me roughly an hour and a half to complete the entire project.

Testing:

For the testing of the program, there is a text file attached to this submission that has different test cases logged. To test the program I went through the program and tested a couple of different use cases. I tested the normal case and I also tested some special use cases to make sure that the error handling was being handled properly.

Conclusions:

Overall the program seems to be able to deal with the typical use cases and can react to them pretty well. The error checking that is implemented is effective and meets the guidelines set in the original assignment. The programming project was a success because it achieved everything it need to and the code was relatively clean and well commented. If I could do something different next time I would use a more object oriented approach. While this assignment was designed to improve our abilities with arrays, I feel that classes would have made this assignment alot more intuitive. The project, including the documentation, took roughly 2 and a half hours to complete.