CS 161A: Programming and Problem Solving I

Assignment 6: Vending Machine

Academic Integrity

You may NOT, under any circumstances, begin a programming assignment by looking for completed code on StackOverflow or Chegg or any such website, which you can claim as your own. Please check out the Student Code of Conduct at PCC.

The only way to learn to code is to do it yourself. The assignments will be built from examples during the lectures, so ask for clarification during class if something seems confusing. If you start with code from another source and just change the variable names or other content to make it look original, you will receive a zero on the assignment.

I may ask you to explain your assignment verbally. If you cannot satisfactorily explain what your code does, and answer questions about why you wrote it in a particular way, then you should also expect a zero.

Introduction

A Vending Machine sells Coffee and Tea for $0.25. You can buy coffee or tea, enter the quantity and get your balance after the machine has dispensed your drink.

Purpose

In this assignment you will write a program for a Vending Machine that sells Coffee and Tea. Your program will first accept coins from the user until the user enters 0 (5, 10, or 25 only - no other numbers), then print a menu for dispensing Coffee or Tea to the user to choose from and then a prompt for the quantity. If the amount deposited is sufficient, the machine will dispense the item(s), else it should loop back and ask for more money, and present the menu again.

After completing this assignment you will be able to:

- Use a do while loop to read coins from the user
- Use while loops to validate user input
- Use an accumulator variable in a loop to keep track of the total amount to pay including the extras

Task

- ❏ Before you get started:
  - ❏ Check out Sample Assignment A06 - Algorithmic Design document
  - ❏ Check out Sample Assignment A06 - Sample6 Code
- ❏ Open the Algorithmic Design Document, make a copy, and follow the steps to create your algorithm.
- ❏ You must express your algorithm as **pseudocode** or a **flowchart.**
- ❏ You will write a program to do the following:
  - ❏ Read coins from the user until the user enters 0. Accept only 5, 10 or 25 for numbers (nickels, dimes, and quarters). Do data validation and stay in a loop if other numbers are entered. Use 0 to end. Keep an accumulator to count the amount of money entered.
  - ❏ Then present a menu to the user (see Sample Run below) for Coffee, Tea or Quit. If the user chooses Quit, output the balance, and quit the program. Your program must use upper and lower case letters. Cost of coffee and tea is $0.25 each.
  - ❏ If the input is not valid, print an error message and prompt again. Continue to give an error message and prompt until the user enters a valid choice. Please see Sample6 Code for all data validation examples.
  - ❏ Prompt the user for the quantity and validate. If the input is not valid, print an error message and prompt again. Continue to give an error message and prompt until the user enters a valid choice. See Sample Run below.
  - ❏ If you use characters, you must accept both uppercase and lowercase.
  - ❏ Print the item they chose and the price of the item. Please create constants to hold the cost of each item.
  - ❏ Multiply the cost of coffee or tea by the quantity and get the total cost.

- ❑ If the deposited money is sufficient, print the total and the balance and end the program.
- ❑ If the deposited money is not sufficient, then go back to the coin collecting loop and ask for more money. ***Remember how much they had deposited before, don't start the collection all over again.*** Then display the menu and ask for quantity again and go through the steps to get the new total, and repeat the process.
- ❑ **Do not use any goto statements. You may not use any breaks, continue, or return statements inside any loops - you are allowed to use breaks inside a switch statement.** Please check the coding guidelines in the style guidelines
- ❑ **Use only concepts you have learned so far.**
- ❑ I have some Sample Runs below - please check all of them to see how your program should behave.
- ❑ Print a goodbye message.

Criteria for Success

- ❑ Test your program using the following sample runs, making sure you get the same output when using the given inputs (in **`blue`**):

```
Welcome to my Coffee/Tea Vending Machine!

Enter coins - 5, 10, or 25 only: 5
Enter coins - 5, 10, or 25 only: 25
Enter coins - 5, 10, or 25 only: 25
Enter coins - 5, 10, or 25 only: 5
Enter coins - 5, 10, or 25 only: 10
Enter coins - 5, 10, or 25 only: 10
Enter coins - 5, 10, or 25 only: 0

Your balance is $0.80

Please pick an option ($0.25 each):
    C/c: Coffee
    T/t: Tea
    Q/q: Quit
>> k
Invalid Option! Please choose a valid option!
>> 9
Invalid Option! Please choose a valid option!
>> c
How many would you like?
>> f
Invalid Option!
How many would you like?
>> 2

Your total: $0.50
Your balance: $0.30

Thank you for using my Vending Machine Program!
```

```
Welcome to my Coffee/Tea Vending Machine!

Enter coins - 5, 10, or 25 only: 5
Enter coins - 5, 10, or 25 only: 25
Enter coins - 5, 10, or 25 only: 0

Your balance is $0.30

Please pick an option ($0.25 each):
    C/c: Coffee
    T/t: Tea
    Q/q: Quit
>> c

How many would you like?
>> 2

Your total is $0.50
Your balance is $0.30
Not enough change!! Please add more coins.

Enter coins - 5, 10, or 25 only: 5
Enter coins - 5, 10, or 25 only: 25
Enter coins - 5, 10, or 25 only: 0

Your balance: $0.60

Please pick an option ($0.25 each):
    C/c: Coffee
    T/t: Tea
    Q/q: Quit
>> T

How many would you like?
>> 1

Your total is $0.25
Your balance is $0.35

Thank you for using my Vending Machine Program!
```

```
Welcome to my Coffee/Tea Vending Machine!

Enter coins - 5, 10, or 25 only: 5
Enter coins - 5, 10, or 25 only: 25
Enter coins - 5, 10, or 25 only: 0
```

```
Your balance is $0.30

Please pick an option ($0.25 each):
    C/c: Coffee
    T/t: Tea
    Q/q: Quit
>> q

Your total is $0
Your balance is $0.30

Thank you for using my Vending Machine Program!
```

- ❏ Check out Sample Assignment A06 - Algorithmic Design document
- ❏ Check out Sample Assignment A06 - Sample6 Code
- ❏ Complete zyBooks section **CS161A 6. Loops Part II** activities.
- ❏ Complete all sections of your Algorithmic Design Document.
- ❏ Include **pseudocode** or a **flowchart** in part d of the design document.
- ❏ **Follow these Coding Construct Requirements:**
    - ❏ Do not use arrays or any vectors for this program. Use only the concepts we have learned so far.
    - ❏ Do not use any goto statements. You may not use any breaks, continue, or return statements inside any loops - you are allowed to use breaks inside a switch statement.
    - ❏ The denomination for the coins (5, 10, 25) must be validated.
    - ❏ All menu options must be checked for valid options - If the input is not valid, print an error message and prompt again. Continue to give an error message and prompt until the user enters a valid choice.
    - ❏ Must print the total and balance and format money to 2 decimal places, i.e., print $0.50 and not 0.5
    - ❏ Must loop back if there is insufficient money, and ask for more money.
    - ❏ Print a welcome and goodbye message.
- ❏ Please open and compare your work with the grading rubric before submitting.
- ❏ Remember to follow all style guidelines.
- ❏ You must have comments in your code per style guidelines.
- ❏ Download your Algorithmic Design Document as a PDF (File -> Download -> PDF), rename it to `a06.pdf`, and upload it to the D2L assignment by **Wednesday**.
- ❏ Upload your `a06.cpp` C++ source file to the D2L assignment by **Sunday**.
- ❏ Do your own work. Consult the syllabus for more information about academic integrity.

Additional Support

- ❏ Check out Sample Assignment A06 - Algorithmic Design document
- ❏ Check out Sample Assignment A06 - Sample6 Code
- ❏ Post a question for the instructor in the Ask Questions! area of the Course Lobby.