CS 31 Project 2

Coffee Break C++

Time due: 9:00 PM, Monday, 1/26/2025

Introduction

Your task is to implement a program to calculate the total charge for a customer at JavaTime Cafe based on their coffee order. The program must validate inputs, compute the charge based on the provided pricing scheme, and handle specific discounts if applicable.
This assignment helps you practice:
1. C++ basics, such as variables, conditional statements, and formatted output.
2. Input validation techniques.
3. Proper adherence to input/output specifications.

Program Specifications

Your program must collect the information for one customer in the order indicated below.
After processing the inputs:
1. Write to 'cout' a line containing only three hyphens ('---').
2. Follow this with one single line displaying the calculated charge or an error message.

Output Requirements

The line after '---' must adhere to one of the following formats. Our grading tool will check only this line (and verify there are no additional lines).

Valid Cases:
For valid inputs:
The total charge for [customer] is $[amount]

Where:
- [customer]: The exact first name as entered by the user.
- [amount]: The total charge, shown as a nonnegative number with at least one digit to the left of the decimal point and exactly two digits to the right (e.g., '$11.86').

Invalid Cases:
For invalid inputs, display the appropriate error message and terminate. The possible error messages are:
1. If the first name does not start with an uppercase letter:
   The first name must start with an uppercase letter.
2. If the number of cups is not positive:
   The number of cups must be positive.
3. If the type of coffee is not 'r' or 'd':
   You must enter r or d.

4. If the extra items input is not ‘y’ or ‘n’:
   You must enter y or n.
5. If the day of the week is not valid:
   You must enter a valid day of the week.

Program Behavior and Grading Guidelines

Your program must output a line with exactly three hyphens (‘---’) before any result or error message. Do not write additional lines (e.g., comments or unnecessary prompts) after the ‘---’ line.Your program must handle input for one customer only and terminate after writing the output.If there are multiple errors, display only the error for the first invalid input in the order inputs are collected.

Common Errors:
- Incorrect Message Format (e.g., Missing period, wrong capitalization, extra spaces, missing dollar sign or decimal digits).
- Extra Output Lines (e.g., gratuitous comments like 'Thank you for choosing JavaTime Cafe!').

Input Requirements

Your program must gather inputs in the following order:
1. Customer's First Name: Must not be empty and the first letter must be uppercase.
2. Number of Cups: Must be a positive integer.
3. Type of Coffee: Must be ‘r’ (regular) or ‘d’ (decaf), in lowercase.
4. Extra Items: Must be ‘y’ (yes) or ‘n’ (no), in lowercase.
5. Day of the Week: Must be a valid day name (e.g., Monday, Tuesday, ..., Sunday).

Pricing Scheme

1. Base Price:
- Regular coffee (‘r’): $4.00 per cup.
- Decaf coffee (‘d’): $4.50 per cup.
2. Extra Items:
- If the customer opts for extra items (e.g., snacks), a flat charge of $1.95 is added to the total.
3. Discount:
- If the day of the week is ‘Wednesday’, the customer receives a 15% discount on the total charge.

Example Dialogs

Valid Inputs:

Input:
Customer's first name: Maria
Number of cups: 3
Type of coffee (r=regular, d=decaf): r

Any extra items? (y/n): y
Day of the week: Wednesday

Output:

The total charge for Maria is $11.86

Input:
Customer's first name: John
Number of cups: 1
Type of coffee (r=regular, d=decaf): d
Any extra items? (y/n): n
Day of the week: Friday

Output:

The total charge for John is $4.50

Invalid Inputs:

Input:
Customer's first name: maria
Number of cups: 3
Type of coffee (r=regular, d=decaf): r
Any extra items? (y/n): y
Day of the week: Monday

Output:

The first name must start with an uppercase letter.

Input:
Customer's first name: Maria
Number of cups: -2
Type of coffee (r=regular, d=decaf): r
Any extra items? (y/n): y
Day of the week: Monday

Output:

The number of cups must be positive.

Restrictions

1. No Loops: Do not use ‘for’, ‘while’, or ‘do-while’.
2. No Arrays or Switch Statements.

3. No Re-Prompting: Handle input for one customer only. Do not re-prompt after invalid input.
4. Ensure your program adheres to the format requirements.

Submission Requirements

Submit a zip file containing:
1. Source Code:
- File name: 'coffeebill.cpp'.
- Include clear, helpful comments in your code.
2. Report:
- File name: 'report.docx' or 'report.txt'.
- Include:
  - Notable obstacles and how you overcame them.
  - A list of test cases used to test your program, along with reasons for each test.

Grading Notes

Your program will be graded based on:
1. Adherence to input/output format specifications.
2. Correctness of computations.
3. Validation of all input types.
4. Handling of edge cases.