CS-181 Lab 3

Student Learning Outcomes

- More experience with the Microsoft Visual Studio Integrated Development Environment (IDE)
- Creating C++ projects in MS Visual Studio
- Using the code editor
- Sequential code
- The cin object
- Arithmetic expressions
- Formatting output

Overview

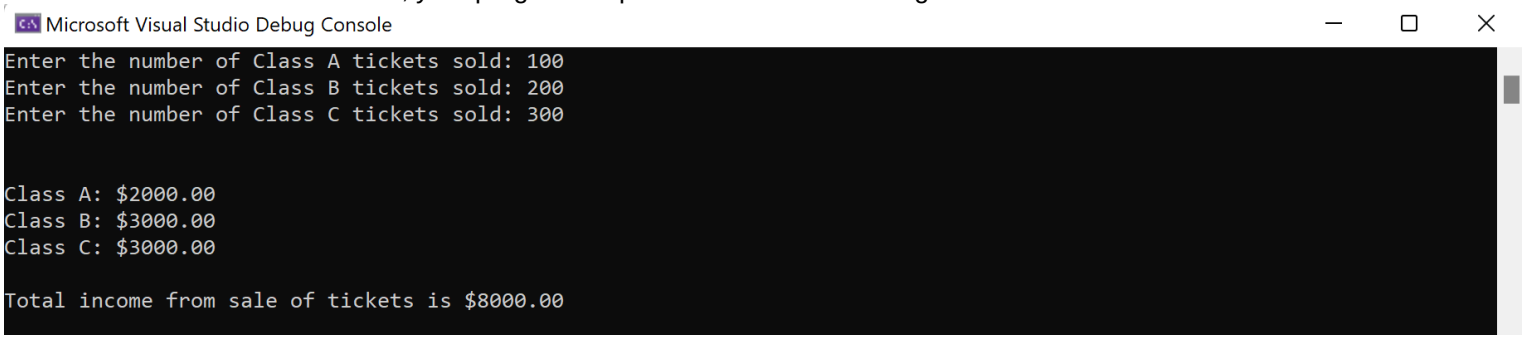
This lab provides a hands-on opportunity to apply some of the terminology and concepts presented in chapter 3. Starting with this lab, you will simply be given the problem statements.

Exercise 1: Stadium Seating

There are three seating categories at a stadium. For a softball game, Class A seats cost $20, Class B seats cost $15, and Class C seats cost $10. Write a program that asks how many tickets for each class of seats were sold, then displays the amount of revenue generated from each seating class and total ticket sales. Format your dollar amount in fixed-point notation, with two decimal places of precision, and be sure the decimal point is always displayed.

Video link to help with developing the solution.

While there is room to be creative, your program output should look something like this:



Be sure to format the total.

Exercise 2: Recipe Adjuster

A cookie recipe calls for the following ingredients:
1.5 cups sugar
1 cup butter
2.75 cups flour

The recipe produces 48 cookies with the amounts of the ingredients listed above.

Write a program that asks the user to input how many cookies they want to make and displays the adjusted amount of sugar, butter and flour needed to make the specified number of cookies.

Hint: For each ingredient, adjusted amount = (number of cookies * recipe amount) / recipe number of cookies

While there is room to be creative, your output should look (roughly) something like this:

This exercise does ***not*** require any special number formatting.

Testing hint: The number of cookies produced by the original recipe is 48. A quick and easy way to make sure your program is working correctly is to test with 24 cookies as shown in the sample program output above. The ingredient amounts should be one half the original recipe amounts. Try testing with twice the number of cookies (96). Ingredient amounts should be twice the original recipe ingredient amounts.

Exercise 3: Fahrenheit to Celsius Conversion

Write a program that prompts the user for a temperature in degrees Fahrenheit, converts the temperature to degrees Celsius, and displays it. The formula is C = (F – 32) * (5.0 / 9.0) where F is the temperature in Fahrenheit and C is the temperature in Celsius.

Your output should look something like this:

This exercise does ***not*** require any special number formatting but be careful with the data types!

Exercise 4: Currency conversion

Write a program that will convert U.S. dollar amounts to Japanese yen and to euros, storing the conversion factors in the constants YEN_PER_DOLLAR and EUROS_PER_DOLLAR. Use the following exchange rates:

1 Dollar = 98.93 Yen

1 Dollar = 0.74 Euros

Your program output should look something like this:

Be sure to format your currency amounts in fixed-point notation, with two decimal places of precision, and be sure the decimal point is always displayed.

Assignment Submission

Your completed assignment must be uploaded to Canvas in a zip file format. Be sure you have folders for all 4 exercises in your Lab03 folder and you have all files and sub-folders needed to open with Visual Studio.

Grading Criteria:

Deliverable	**Points**	**Breakdown**
Exercise 1	13	Opens in VS, code is clear (I.e., appropriate variable names, use of constants, comments, program header), compiles, runs, produces correct output. Output formatted correctly
Exercise 2	12	Opens in VS, code is clear (I.e., appropriate variable names, use of constants, comments, program header), compiles, runs, produces correct output.
Exercise 3	12	Opens in VS, code is clear (I.e., appropriate variable names, use of constants, comments, program header), compiles, runs, produces correct output.
Exercise 4	13	Opens in VS, code is clear (I.e., appropriate variable names, use of constants, comments, program header), compiles, runs, produces correct output. Output formatted correctly
Lab Total	**50**	