Programming Fundamentals
CS1336

Assignment #5

Assignment #5 – Selection Logic

Introduction

Your fifth programming assignment will consist of two small C++ programs. Each program will be independent of the other program. Each one should compile correctly and produce the specified output.

Please note that each of the programs should comply with the commenting and formatting rules we discussed in class. For example, there should be a header for the whole program that gives the author's name, class name, date, and description. End braces should be commented, and there are alignment and indenting requirements as discussed. Please ask if you have any questions.

Program #1

For an introduction to selection logic, please implement Problem 7 on page 223 (9E) of our book.

Your program should prompt the user for some number of seconds. It will then determine if the number of seconds entered is (a) less than 60 seconds, in which case it will report in seconds, (b) between 60 seconds but less than 3,600 seconds, in which case it will print a report in minutes, (c) between 3,600 seconds but less than 86,400 seconds, in which case it will print a report in hours, or (d) greater than or equal to 86,400 seconds, in which case it will print a report in days.

An *if-else-if* sequence (Gaddis, p. 178, 9E) would be a natural solution to this problem. Also, for the moment, you do not have to worry about input validation (e.g., checking to make sure the number is positive).

Some example runs might look like:

```
Please enter the number of seconds:     45
45 seconds is 45.00 seconds.

Please enter the number of seconds:     1234
1234 seconds is 20.57 minutes.

Please enter the number of seconds:     52134
52134 seconds is 14.48 hours.

Please enter the number of seconds:     100000
```

```
100000 seconds is 1.16 days.
```
Note that all decimal numbers are printed to two decimal places.

A scan of the problem follows.

> **7. Time Calculator**
>
> Write a program that asks the user to enter a number of seconds.
>
> - There are 60 seconds in a minute. If the number of seconds entered by the user is greater than or equal to 60, the program should display the number of minutes in that many seconds.
> - There are 3,600 seconds in an hour. If the number of seconds entered by the user is greater than or equal to 3,600, the program should display the number of hours in that many seconds.
> - There are 86,400 seconds in a day. If the number of seconds entered by the user is greater than or equal to 86,400, the program should display the number of days in that many seconds.

Program #2

Please implement a modification of Problem 25 on page 228 (9E). A scan of the original problem as presented in the book is provided below.

Note that this problem requires the implementation of a small menu system (see Section 4.10) as well as input validation (see Section 4.11). The menu system can be implemented at our current level with an "`if-else-if`" ladder or by a `switch` statement. The input validation can be done with standard "if" statements.

Program 4-18 is an excellent model for the solution to this problem, at least as far as the basic structure is concerned. Make sure you understand exactly what is to be calculated in each "package." For example, if Package A is chosen and the user wants 650 minutes, how should the monthly bill be calculated exactly? I suggest you work out those equations in pseudo-code before you begin writing the program.

Also, take note of the input validation requirements at the end of the problem (in italics). They specify that only 'A', 'B', or 'C' is allowed as the input for the package. Have your program accept either upper or lower case letters for the package choice (e.g., either 'A' or 'a' will choose package 'A'). Anything outside of the range 'A' – 'C' or the range 'a' – 'c' should be considered an error.

Also, note that the problem asks the user to enter how many minutes were used even if Package "C" is chosen. This may seem redundant in the case of Package "C," but it will be useful in the following modification. Consequently, your program should ask both of those questions up front, i.e., the package and the number of minutes, before it enters the `if-else-if` structure to do the processing on the package.

Bonus Points

The basic problem asks you to get the package and minutes information from the user and then print out what the cost to the user would be. If you successfully complete this project, you will get 100%.

However, for 5% bonus points, let's modify the problem in the following way to make it more useful. Have your program calculate the monthly bill for all three packages given the number of minutes entered by the user. If the user chooses the package with the best (i.e., the lowest) monthly rate, report his monthly bill but also print a message congratulating him on making the right choice. If the user chooses a different package than the best one, then report the monthly bill for the package he chose but also inform him that he had a better option available and what the best rate and package would have been.

The bonus section requires you to (a) determine the cost for all three packages, (b) determine the lowest cost package, (c) compare it against the user's choice. It's a nice selection logic problem, and there is pseudocode for it on eLearning. In addition, there is a new post on eLearning describing how to calculate the minimum value of three numbers.

25. Mobile Service Provider

A mobile phone service provider has three different subscription packages for its customers:

Package A: For $39.99 per month 450 minutes are provided. Additional minutes are $0.45 per minute.

Package B: For $59.99 per month 900 minutes are provided. Additional minutes are $0.40 per minute.

Package C: For $69.99 per month unlimited minutes provided.

Write a program that calculates a customer's monthly bill. It should ask which package the customer has purchased and how many minutes were used. It should then display the total amount due.

Input Validation: Be sure the user only selects package A, B, or C.