CSCE 1030 – Homework 2

Due: 11:59 PM on Wednesday, February 25, 2015 CST

Problem Statement:

The purpose of this programming project is to write a small C++ program to perform a simple arithmetic operation based on input from the user.

Your program's output should initially display the department and course number, your name, your EUID, and your e-mail address.

You will first prompt the user to enter an integer between –10 and 10, inclusively. If the number is not in this range, you will display a meaningful error message before re-prompting the user to enter the integer again. Specifically, you must display the integer entered and indicate whether the integer that the user entered is less than –10 or greater than 10. For example, if the user entered –12, the error message should display "The number you entered (–12) is less than –10."

If the user entered a valid integer, you will then display the integer itself as part of a meaningful message indicating whether the number was positive, negative, or zero. For example, you may display "You entered a positive number (9)." when the user entered the integer 9.

You will then prompt the user to enter any other integer (without restrictions, except to be in range for the `int` data type). Following, you will prompt the user to enter the type of arithmetic operation he/she wishes to perform: addition (+), subtraction (–), multiplication (*), or division (/). If the user enters an operation other than those listed (+, –, *, or /), you will default to the addition (+) operator.

You will perform the arithmetic operation as indicated by the user, showing the details of the operation as well as the result in a meaningful message displayed to the screen. A place to be careful here is if the user enters 0 for the second number and requests a division operation. In this case, you will handle the division by zero case by displaying a meaningful error message and terminating the program.

You may assume that all input will be of the appropriate data type, although the range may not be valid.

Design:

On a piece of paper (or word processor), write down the algorithm, or sequence of steps, you will use to solve the problem. You may think of this as a "recipe" for someone else to follow. Continue to refine your "recipe" until it is clear and deterministically solves the problem. Be sure to include the steps for prompting for input, performing calculations, and displaying output.

Type these steps into a document (Word, txt, PDF, etc.). Note that this should be done before you start coding as completing it afterwards does not help you in learning the design process.

Implementation:

Now that you have a working design, your next step is to translate these steps into C++ code. Use the algorithm development techniques discussed in class to implement your

solution to the problem above. Add your C++ code a little at a time, and compile and test as you go.

Remember to add your comments to your code to explain your program. Do this before/during programming instead of waiting until the end. At a minimum, you should comment the header (e.g., name, class, date, brief description of the program, etc.), all variables (i.e., what they are used for), and specific "blocks" of code. For example, use comments to describe the inputs, the formulas used, and any other important steps, such as loops, in your code.

Your program will be graded based largely upon whether it works correctly on a CSE Department machine, so you should make sure your program compiles and runs on a CSE machine.

Your program will also be graded based upon your programming style. At the very least, your program should include:

- A consistent indentation style as recommended in the textbook and in class;
- Meaningful variable names;
- A block header comment section that includes: your name, e-mail address, and a brief description of the program.

Testing:

Test your program to check that it operates as desired with a variety of inputs. Compare the answers your code gives with the ones you get from hand-calculations. Although your program is not required to check for incorrect inputs, observe the effect of such inputs. Try typing "hello world" when your program asks for a number. What is the result?

Sample input and output appears below (with input shown in **bold**):

```
+-----------------------------------------+
|       Computer Science and Engineering       |
|        CSCE 1030 - Computer Science I        |
|   Student Name     EUID    euid@my.unt.edu   |
+-----------------------------------------+

Please enter an integer in the range -10 to 10: -12
The number you entered (-12) is less than -10.
Please enter an integer in the range -10 to 10: 43
The number you entered (43) is greater than 10.
Please enter an integer in the range -10 to 10: 9
You entered a positive number (9).
Now enter any other integer: 4
Enter the arithmetic operation (+, -, *, or /) you would like to
perform: /
Integer Division: 9 / 4 = 2
```

CSCE 1030 – Homework 2

Due: 11:59 PM on Wednesday, February 25, 2015 CST

Another sample input is as follows (with input again shown in **bold**):

```
+-----------------------------------------+
|       Computer Science and Engineering       |
|        CSCE 1030 – Computer Science I        |
|   Student Name     EUID    euid@my.unt.edu   |
+-----------------------------------------+

Please enter an integer in the range -10 to 10: -6
You entered a negative number (-6).
Now enter any other integer: 0
Enter the arithmetic operation (+, -, *, or /) you would like to
perform: /
This operation is not valid: Division by Zero!
```

Documentation:

When you have completed your C++ program, write a short report (2 – 3 paragraphs) describing what the objectives were, what you did to solve the problem, and the status of the program. Does it work properly for all test cases? Are there any known problems? Also include a reflection in what you learned by completing this program that you anticipate taking forward as your knowledge in C++ programming grows.

Save this report in a separate file to be submitted electronically. You should also include any specific instructions required to compile or execute your code.

Homework Submission:

In this class, we will be using electronic homework submission to make sure that all students hand their programming projects (and labs) on time. You will submit your program source file to the class website through the "**Homework 2**" drop box by the due date and time.

Note that this project must be done individually. The program will be checked using a code plagiarism tool against other solutions, so please ensure that all work submitted is your own.

Note that the dates on your electronic submission will be used to verify that you met the due date above. All homework up to 24 hours late will receive a 50% grade penalty. Later submissions will receive zero credit, so hand in your best effort on the due date.

Summary:

- You will design an algorithm (or steps used) to solve the problem.
- You will implement your program on the CSE machines using C++. You will make sure to use good style, good variable names, indentation, etc. You will compile, run, and test your code.
- You will write a brief report describing what your code does and how well it works.
- You will submit electronically your C++ code, your design, and your brief report.