

ECE 71 – Introduction to Computational Programming

Professor Kriehn – Fall 2017

Code Due By: Midnight on Tuesday, September 26, 2017

Writeup Due By: Class on Wed/Thu, September 27/28, 2017

HOMEWORK #11 – Function Overloading

Write an overloaded function `max()` that takes either two or three parameters of type `double` and returns the largest value.

Specifications:

Use two functions called `max()` that are overloaded to solve the problem. The program should prompt for 2 inputs, call the appropriate `max()` function, and print the maximum value. The second part of the program should prompt for 3 inputs, call the appropriate `max()` function, and again, print the maximum value. Print the results with two digits after the decimal point.

As an example, if you execute the program with the following underlined inputs, the output will be:

```
~> main.o
Enter x: 9.7
Enter y: -4.5

The maximum value is: 9.70

Enter x: 13.2
Enter y: -23.4
Enter z: 44.2

The maximum value is 44.20
~>
```

Develop your I/O diagram and pseudocode, debug your code, and submit to the Grader Program.

ECE 71 – Introduction to Computational Programming

Professor Kriehn – Fall 2017

HOMEWORK #12 – Rounding to a Specific Decimal Place

An application of the `floor()` function is rounding a value to a specific decimal point. The statement

```
y = floor(x + 0.5);
```

rounds the variable `x` to the nearest integer and assigns the result to `y`. The statement

```
y = floor(x * 10.0 + 0.5)/10.0;
```

rounds the variable `x` to the tenths place and assigns the result to `y`. The statement

```
y = floor(x * 100.0 + 0.5)/100.0;
```

rounds the variable `x` to the hundredths place and assigns the result to `y`.

Write a program that defines four functions to round a variable `x` various ways with the following function prototypes:

- a) `int roundToInteger(double x);`
- b) `double roundToTenths(double x);`
- c) `double roundToHundredths(double x);`
- d) `double roundToThousandths(double x);`

Use these functions to round the average of a sequence of numbers to the proper decimal point.

Specifications:

Prompt the user to enter the number of values to be averaged. Use this value to create a `for()` loop to read in a sequence of numbers. Calculate the average of the numbers, and pass the average into each of the functions to round the value to the required decimal points. Print the results to the screen with 4 values after the decimal point, as necessary.

As an example, if you execute the program with the following underlined inputs, the output will be:

```
~> main.o
Enter Number of Values to be Averaged: 3

Enter a number: 1.299999
Enter a number: 4.5321
Enter a number: -2.3

The raw average is:          1.1774
...to the nearest integer:   1
...to the tenths place:     1.2000
...to the hundredths place: 1.1800
```

...to the thousandths place: 1.1770
~>

Develop your I/O diagram and pseudocode, debug your code, and submit to the Grader Program.