CSSSKL142 - Lab2

Arithmetic, non-void methods and keyboard input

University of Washington Bothell

Create a class called Lab2a for parts 1-3 and another class called Lab2b for part 4, or you may use the provided templates.

Part 1 (*Lab2a*)

a. In your **Lab2a,** code the exercise you just completed in class in main() and print the variable values to the screen to compare your answers with what your program output.

b.

```
//what is result1?
int result1 = (7 * 3) % 4 - (2 * 6) / 5 + 10 / (3 / 2);

//what is result2?
int result2 = 2 / (4 - 3) + ((5 * 4 % 3 * 5 - 6) * 2) - 5;

//what is k?
int i = 10, j = 20, k; k = ++i - j--;
```

Part 2

(*Lab2a***)** In this exercise you will:

a. Write a method called **kelvins** that takes as input (that is, as **argument**) a temperature in degrees Fahrenheit (of type double) and **returns** the temperature in degrees Kelvin (also a double). Here is the mathematical formula for conversion:

$$K = \frac{5}{9}(F - 32) + 273.15$$

b. Write another method called **temperaturePrinter** that takes as input (i.e. as argument) two doubles and prints the following message to the screen:

```
<double1> F corresponds to <double2> K
```

where <double1> is the temperature in degrees Fahrenheit and <double2> is the corresponding temperature in degrees Kelvin.

c. You should test your methods **kelvins** and **temperaturePrinter** in the main method by calling them with 5 different temperatures in degrees Fahrenheit to be converted to Kelvins.

Note: you will have to call both of your methods for each temperature you pick. Also, try to pick numbers that are easy for you to check, such as 32 F.

Part 3

(*Lab2a***)** In this exercise you will:

a. Write a method called **secondTime** that takes as argument an integer corresponding to a number of seconds, computes the exact time in hours, minutes and seconds, then prints the following message to the screen:

```
<inputseconds> seconds corresponds to:
<hour> hours, <minute> minutes and <second> seconds
```

b. Write another method called **inSeconds** that takes as arguments three integers: hours, minutes and seconds, computes the exact time in seconds, then **returns** the total number of seconds and **prints** the following message to the screen:

```
<hours> hours, <minutes> minutes and <seconds> seconds corresponds to:
<totalseconds> seconds
```

c. Write a few tests (5) for the **inSeconds** method. At least one of your tests should use the return value and one should simply use the method as if it were a void method.

d. Test your **secondTime** method (from part **a** above) in the main method using keyboard input. For this purpose, make sure that before you use any Scanner methods, you print a message to the screen prompting the user of your program to enter the appropriate input. In this case, you message will ask for a total number of seconds to be entered as an integer.

Part 4

(***Lab2b***) Make sure that you are done with Lab2a before you move on to this part. In this exercise you will:

a. Use two methods from the Math library, namely **max** and **min**. Here is a quick introduction to these methods: they each take two numeric inputs (i.e. arguments) and return whichever number is the maximum or minimum (depending on if you are using max or min) of the two.

Let's try them out! In main, declare and initialize variables as follows:

```
double maxNumber1 = Math.max(22.3, 34.5);
double minNumber1 = Math.min(3.6/7.2, 3.8/6.9);
double maxNumber2 = Math.max(2/3, 0.1);
double minNumber2 = Math.min(13.5555, 13.5556);
```

What are their values? Write them down somewhere. Now print them to the screen, did you get what you expected? If not, reflect on the reason and ask someone. Make sure you understand how to use these methods before proceeding.

b. Write a program in main that asks the user of your program to first enter his or her name followed by 3 real numbers (that is, any number from $-\infty$ to ∞). Your program should sort the three numbers in descending order, then print to the screen a greeting to the person and the sorted numbers. For example **(your input in blue)**,

```
Please enter you name followed by three numbers (space separated):
Leonardo -22.3 12.2 2.3
Hi there, Leonardo! Here are the numbers you entered in descending order:
12.2 2.3 and -22.3
Thank you for using the three-number-sorting system! Good-bye.
```

Note: You must use the **Math.max** and **Math.min** methods introduced above. The class has not covered conditionals, so please avoid using conditionals!

Make sure to run your program several times and test it with several inputs of different orderings.

Part 5

Submit your both **Lab2a.java** and **Lab2b.java** to Canvas by due date.