COSC 1436: Programming Fundamentals I
Lab 09: File I/O and Methods

Please submit one source code file (.java file) for each exercise. (Ex. Lab5Ex1.java, Lab5Ex2,java, Lab5Ex3.java ...etc).

Remember to follow the commenting guidelines.

Program 1: Unit conversion (20 points):

In this program, you will read a list of numbers from a file and write numbers into a different file. On Blackboard, there is a file that you will use for the input for this program called " Lab09Input1.txt". Download it and save it in the same folder as this BlueJ program. This file contains a list of lengths in inches. Open the file by double clicking on it to see what the format looks like.

Your program will read the next length from the input file, convert it to centimeters, and then write it to a file named "metric.txt". The program will repeat this process (read from input file, convert, write to output file) until it reaches the end of the input file, print “Done” on the screen, and then quit.

Your output file should have the following values in it:

```
10.668000000000001
9.144
25.4
17.526
3.302
18.796000000000003
```

Program 2: Multi-conversion (30 points):

Write a program that converts English measurements to metric measurements for pounds to kg, miles to km, and gallons to liters. You may call the methods whatever you’d like.

Your program will have 3 methods:

- The first method will convert pounds to kg, by accepting an integer number of pounds as an input argument, doing the conversion kg=pounds/2.2, and returning the number of kg as a double.
- The second method will convert miles to km, by accepting an integer number of miles as an input argument, doing the conversion km=1.61*miles, and returning the number of km as a double.

- The third method will convert gallons to liters, by accepting an integer number of gallons as an input argument, doing the conversion liters=3.79*gallons, and returning the number of liters as a double.

Your main program will ask the user to enter a quantity to convert (**integer**) and the units (a **char** value).

- If the units is a '`p`', then the program will call the pounds to kg method and send the quantity as the number of pounds. After the result comes back from the method, it will print the number of pounds and the equivalent number of kg.
- If the units is an '`m`', then the program will call the miles to km method and send the quantity as the number of miles. After the result comes back from the method, it will print the number of miles and the equivalent number of km.
- If the units is a '`g`', then the program will call the gallons to liters method and send the quantity as the number of liters. After the result comes back from the method, it will print the number of gallons and the equivalent number of liters.

Example:

```
Please enter an integer quantity and the units:

10

g

10 gallons is equal to 37.9 liters
```

Hint: Instead of using `nextInt()`, use `Integer.parseInt(keyboard.nextLine())` This is another one of those cases where `nextInt()` has trouble with return chars.

Program 3: Rock, Paper, Scissors (30 points):

Write a program that lets the user play the game of Rock, Paper, Scissors against the computer. The program should work as follows.

Your program will have 4 methods:

- The first method. When the program begins, the computer will automatically generate a random number in the range of 1 through 3. This random number will represent it's choice. If the number is 1, the computer has chosen rock. If the number is 2, the computer has chosen paper. If the number is 3, the computer has chosen scissors. This will return the CPU choice.

- The second method. The user enters his or her choice of rock, paper, or scissors at the keyboard. (You can create a nifty menu to display the options). This method will return the human choice. **Provide input checking so the user doesn't choose an incorrect choice.**

  Menu Example:

  ```
  1. Enter 1 for rock
  2. Enter 2 for paper
  3. Enter 3 for scissors
  ```

- The third method will display the computer's choice to the console.

- The fourth and final method will grab Human Choice and CPU choice and make a winner decision. A WINNER is selected according to the following rules:
    a. If Player chooses rock and the CPU chooses scissors, then rock wins. (Rock smashes Scissors).
    b. If Player chooses scissors and the CPU chooses paper, then scissors wins. (Scissors cut Paper).
    c. If Player choses paper and the CPU chooses rock, then paper wins. (Paper wraps Rock)
    d. If both Players make the same choice, the game must be played again to determine the winner. (Return true if the game ended in a tie to run the whole game again.)
    e. Make sure to print the appropriate WINNER and LOSER labels for each game.

Multiple Choice Questions (20 points)

Please include your answers to the multiple choice questions in comments at the end of your 3rd program from this lab.

1. Given the following statement, which statement will write "Calvin" to the file DiskFile.txt?

```
PrintWriter diskOut = new PrintWriter("DiskFile.txt");
```

```
a) System.out.println(diskOut, "Calvin");
b) DiskFile.println("Calvin");
c) PrintWriter.println("Calvin");
d) diskOut.println("Calvin");
```

2. When should you use a `.close()`?

a) after every time you read from a file or write to one
b) after you are finished doing all of the reads or writes in a program, but it does not have to be the last line.
c) only after writing to a file
d) only after the last read from a file
e) if you read from a file or write to one, it must always be the last line of the program

3. `inputFile` is a `Scanner` object that was used to open a file. Which of the following `while` loops shows the correct way to read data from the file until the end of the file is reached?

```
a) while (inputFile != null){
   …
   }


b) while (!inputFile.EOF){
   …
   }


c) while (inputFile.hasNext()==true){
   …
   }


d) while (inputFile.nextLine == " "){
        …
        }
```

4. Which of the following statements are true about the differences between a text file (.txt) and a word processor file (like Microsoft Word .docx files)

    I. Text files contain only letters
    II. Text files contain only letters, numbers and symbols
    III. Text files do not have any formatting or fonts

    a) only I
    b) only II
    c) II and III
    d) I and III
    e) I and II

5. What is the output after the execution of the following code:

```
public static void main(String[] args){

        int a = 10, b = 20;

        showValue(b, a);

}

public static void showValue(int a, int b){

        System.out.println("The first number is " + a +

        ", the second number is " + b + ".");

}
```

    a) The first number is 10, the second number is 20.
    b) The first number is 20, the second number is 10.
    c) The first number is 0, the second number is 0.
    d) This code has error and does not run.

6. What is the output after the execution of the following code:

```
public static void main(String[] args){

      int a=0;

      int b = 2;

      method1(b);

}

public static void printChar(int a){

      int result;

      result=a+1;

      System.out.print(result);

}
```

a) 0
b) 1
c) 2
d) 3
e) This code has error and does not run.

7. In the following method declaration, what is the return type?

```
public static int myMethod (String returnValue, double returnThis){
      …
}
```

a) String
b) void
c) double
d) static
e) int

8. In the following method header, what types are the input values (in order)?

```
public static boolean StarWars (String planet, double Anakin){
     …
}
```

a) one integer
b) one double and a boolean
c) a boolean
d) a string and a double
e) a double and a string

9. **True/False**: For call by value, the actual value of local variables is changed.

10. **True/False**: When passing multiple arguments to a method, the order in which the arguments are passed is not important.