CHALLENGE LAB 2
Due date: October 31, 2016

What is this lab about and what will you be working on?

In this lab, you will practice new tools and concepts you've learned in class and in lab since midterm 1. Namely, this lab will be on **methods**, **arrays** (1D and 2D), and on **repetition** (via **loops** and **recursion**).
In this lab, you will have to complete two activities. Here is what you have to do:

Activity 1. [60 points] **Method Luhn.** You will have to implement the ***Luhn Checksum Validation*** algorithm to check whether a credit card number is valid or fake. This is a widely used algorithm that checks that you enter a valid credit card number when you are trying to make a purchase.

> **Short description of the algorithm** [Think Like a Programmer an Introduction to Creative Problem Solving by V. A. Sprau]: Using the original number, double the value of every other digit, starting with the leftmost one, as shown below. Then add the values of the individual digits together (if a doubled value now has two digits, add the digits together – see below). The identification number is valid if the sum is divisible by 10 (i.e., the sum has to be a multiple of 10).

Example of execution of the algorithm:

Suppose you want to check that your credit card number is valid. You credit card number is 8273 1232 7351 0569. Let's see how to check it:

8	2	7	3	1	2	3	2	7	3	5	1	0	5	6	9

First, you are going to double every other number, starting with the first number (here, it is number 8):

8 x 2	2	7 x 2	3	1 x 2	2	3 x 2	2	7 x 2	3	5 x 2	1	0 x 2	5	6 x 2	9

And you obtain:

16	2	14	3	2	2	6	2	14	3	10	1	0	5	12	9

But we do not want double digits, so for every number that now has double digits, we add these digits:

1 + 6	2	1 + 4	3	2	2	6	2	1 + 4	3	1 + 0	1	0	5	1 + 2	9

And we obtain:

7	2	5	3	2	2	6	2	5	3	1	1	0	5	3	9

Now we add all of these numbers:
7 + 2 + 5 + 3 + 2 + 2 + 6 + 2 + 5 + 3 + 1 + 1 + 0 + 5 + 3 + 9 = 56
56 is not a multiple of 10, so the credit card number was a fake…

To complete this activity, you will have to:
1/ Write the pseudocode of Method Luhn in the word document that you will submit along with the java file of this lab;
2/ Implement Method Luhn in the file challenge2.java and add relevant code in the main method to test it; and
3/ Provide two examples of input and the corresponding output for Method Luhn – one valid and one invalid credit card number.

Activity 2. [40 points] **Method checkCustomers.** Now you have a 2D array containing the customers' names and their respective card numbers. You have to use the method of Activity 1 to check whose card number is not valid. Each time you find a customer with an invalid card number, you have to write their information (name and card number) in a file named customers.error.

To complete this activity, you will have to implement two versions of this method: **checkCustomersIter** (iterative version based on the use of loops) and **checkCustomersRec** (recursive version with no loop).
For each of these methods, you have to:
1/ Write the pseudocode of it in the word document that you will submit along with the java file of this lab; and
2/ Implement it in the file challenge2.java and add relevant code in the main method to test it.

Bonus Activity. [30 points] **Method generateCCard.** In this bonus activity, you have to design and implement a method called generateCCard, which generates valid credit card numbers. This method does not take any input and returns a 16-digit valid credit card number. This number should change at each execution of the method (i.e., the generation should be random, yet valid).

If you choose to complete this bonus activity, you will have to:
1/ Write the pseudocode of it in the word document that you will submit along with the java file of this lab; and
2/ Implement it in the file challenge2.java and add relevant code in the main method to test it.

What do you have to turn in?

1/ The updated challenge2.java file with your name as the first line of the code
2/ A full description of the methods of Activities 1 & 2 in a word document: as specified in each activity.

When does this have to be turned in?

As soon as you have completed it (so we can provide timely feedback) and no later than October 31 (at 11:59pm).

How do I turn it in?

On piazza, under folder challenge_lab_2.

What will your grade be made of?

For each method you work on, 60% are assigned to the pseudocode and 40% to the code. For more details, ask your TA.