ECE25100 Object-Oriented Programming
Assignment 1

****

NOTE:

1. Programs *must* compile without error. If your program does not compile, you will lose **50%** of the points. Even if your program is "essentially" correct, you will still lose 50% of the points; there is no reason to turn in a program with syntax errors.
2. To get full credit for an assignment your program must completely solve the stated problem.
3. Your program should contain a reasonable amount of comments. At the very beginning of each file, there should be a comment containing the course number, your name, the assignment number, and the date. It is good practice to insert comments into the source code, as you do when programming in any other languages.

```
/*
Class: ECE25100 Object Oriented Programming
Instructor: Xiaoli Yang
Author: [Your Name]
Assignment: [No.]
File Name:
Date: [MM]/[DD]/[YY]
*/
```

4. You should follow the programming styles and guidelines set forth in class. In particular, use meaningful names for variables. For each coding question you should also write a test class and test your code thoroughly.
5. Compress all .java files into one zip file and name it with your full name firstname_lastname_assignment #.zip. Submit the zip file on blackboard.

****

Question 1:

The **Change Calculator:** You are off to the candy store with a **$5** bill to buy some candies. Write a program called **ChangeCalculator** that allows you to enter the total cost of the candies that you are buying (i.e., use the **Scanner** class) in cents (i.e., 127 cents is $1.27). The program should then tell you how many dollars, quarters, dimes, nickels and pennies that you should get as your change (assuming that you always pay with a **$5** bill). Your program should return the least amount of coins possible. All your calculations should be in terms of cents so that you can use just **int** variables. Please handle exceptions nicely in your program.

Your program should have nice-looking output as follows:
Enter the cost of the candies (in cents): **127**
The change from **$5.00** for **$1.27** of candies is:
3 dollar(s)
2 quarter(s)
2 dime(s)
0 nickel(s)
3 pennie(s)

Notice that amount entered is an integer (i.e., total cents) but the output statement displays the cost as a double (i.e., dollars). Test your code with the following costs:
0, 25, 100, 101, 107, 150, 201, 235, 400, 498, 499, 500, 700

Question 2:

Modify the program from *lab 1* so that it prompts the user for a 5-digit positive integer with any number of digits up to 9. The program should then display their integer as a big banner.

PART A: Modify the below test class `StarDisplayTest( )` so that it can prompt the user to enter a 5-digit positive integer, and pass the integer as a parameter to the method `displayStar(int number)` defined in the class `StarDisplay`. (Note: Do not worry about invalid inputs.)

```
import java.Util.Scanner;
public class StarDisplayTest {
    public static void main(String args[]) {
        int a;
        StarDisplay s = new StarDisplay( );
        //prompt the user to enter a 5-digit positive integer
        //and initialize the above variable a:

        s.displayStar(a);
    }
}
```

PART B: Modify the code below, complete the method `displayStar(int number)` to extract 5 digits and call `displayStarHelper(int i)` to display them as a big banner. (Hint: you can extract these digits using the / and % operation)

PART C: Complete a method `displayStarHelper(int i)` so that it can display a digit passed as the parameter. (Hint: You can use IF statement to display the digit as a banner.)

For example if the user enters 52612 the program should display:

```
**      ********
**      ********
**      **      **
**      **      **
********      **
********      **

********      **
********      **
**      **      **
**      **      **
**      ********
**      ********

**************
**************
```

```
**     **     **
**     **     **
********     **
********     **

**************
**************

********     **
********     **
**     **     **
**     **     **
**     ********
**     ********
```

```
public class StarDisplay {
     public void displayStar(int number){
          //extract each digit from the passed parameter, and
          //call the method displayStarHelper(int i)
     }

     public void displayStarHelper(int i){
          //display a banner based on the passed parameter.

     }
}
```