CSCI 2540 Assignment 1

75 points

Due date: Tuesday, Jan. 25, 11:59pm

****

Instructions for setting up your program structure in Eclipse:

If you have not already created a Java Project called "CSCI2540", create one first.

Right click "CSCI2540" and create a new package for this assignment. Name the package as assg1_YourPirateId.

Right click the package and create a new class for each problem in this assignment.

****

1. In the United States, we commonly express dates by listing the month, then the day then the year, all separated by slash marks. In Europe, however, dates are commonly expressed by listing the day first, then the month, then the year, all separated by periods.

   Write a Java application program (**name the file as DateChange.java**) that will get a line of text containing a date in US form from the user, and then change this date to European form.

   Begin by asking the user to enter a date in the form of month/day/year. Store this date in a String variable.

   Next, use the appropriate String methods to swap the month and day parts of the date, and replace the slash marks with periods. Print the revised String to the screen.

   Please note that users are allowed to input the year in two digits or four digits, and input the month and day in one digit or two digits. Your program should be able to handle all possible cases.

   The following is an example of what you might see on the screen when your program runs. The exact output depends on the values that the user types in while the program runs. The user's values are shown below in italics:

   ```
   Enter a date in the form mon/day/year:
   06/17/2008

   Your date in European form is:
   17.06.2008
   ```

Here is another example run:

```
Enter a date in the form mon/day/year:
5/3/09

Your date in European form is:
3.5.09
```

2. Write a Java application program (**name the file as DigitReverse.java**) that will get multiple non-negative integers from the keyboard, one at a time (-1 to exit). For each integer, it should print the digits in reverse order, one per line. **Your program should include the main method and a method called digitExtract**. You main method should allow user to enter multiple integers (non-negative integers, except for -1). The digitExtract( ) method has an integer as a parameter. The method should break up the integer into individual digits (using some simple math) and print them to the screen in reverse order, with each digit appearing on a separate line.

   For example:

```
Please enter a non-negative integer (-1 to exit): 57321

Your number printed in reverse order is:
1
2
3
7
5

Please enter a non-negative integer (-1 to exit): 0

Your number printed in reverse order is:
0

Please enter a non-negative integer (-1 to exit): -1
```

3. Write a Java program (**name the file as CountGrades.java**) that gets a list of grades from the user and counts the number of grades above the average grade.

   Begin your program by asking the user to enter the number of grades. Store this value in a variable called `numOfGrades`. **If the number is a negative number or zero, you program should print some message and stop.**

Next, create an array of size `numOfGrades`. Then you will read `numOfGrades` values representing student grades from the keyboard and place them into the array.

Next, find the average grade of all the values entered, and print this value to the screen.

Finally, count the grades that are above the average grade. Display each individual grade that is above the average grade, and display the count of such grades.

For example:

```
Please enter the number of grades: 4

Enter a grade: 80
Enter a grade: 65
Enter a grade: 78
Enter a grade: 90

The average grade is: 78.25

The grades above the average grade are:
80
90
for a total of 2 grades
```

****

Technical Notes

- At the top of EVERY Java file you create, you must include a comment which states your name. Include other commends in your programs as needed.
- Programs that do not compile will receive an automatic grade of "F".

Submission instructions:

You need to submit your programs electronically on Canvas.

Please **use a named package for each of your assignment.** For example, for assignment 1, create a new package and **name your package as assg1_yourPirateId** (use lower case for your pirate id), such as assg1_smithj19. You also need to include a statement such as "package assg1_smithj19;" at the beginning of each of your .java file (it will be automatically generated in Eclipse if you create the class inside the given package). **Please follow this naming convention exactly for all future assignments. You will be deducted points for not doing so.**

When you submit your files to Canvas, please submit a zip file with your package folder inside the zip file. The package folder should include only .java files (make sure you include .java files, not .class files). The name of the folder should match with your package name. (You can use 7-zip software to zip files/folder). **Once your zip file is unzipped, it will generate a folder that matches your package name.**

****