Computer Science 111 Intro to Algorithms and Programming: Java

Programming Project #2 – User Input and Selections (30 points)

Due 9/27/18

SumOfDigits (5 points): Create a NetBeans project named '**Project2**'. In Project2, create a main class with the name '**SumOfDigits**'. In the main method, write a program that prompts the user to enter an integer between **0** and **1000** (not including 0 and 1000) and adds all the digits in the integer. For example, if the user enters an integer **932**, the sum of all its digits is 9 + 3 + 2 = **14**.

Hint: Use the **%** operator to extract digits, and use the / operator to remove the extracted digit. For instance **932 % 10 = 2** and **932 / 10 = 93**.

From the Project Navigator you can run the new program by right clicking on the file name and selecting 'Run File' or if the file is active in the editor window press 'Shift - F6'. Here is a sample run in NetBeans:

```
Enter an integer between 0 and 1000: 999 Enter
The sum of all digits in 999 is 27
```

AverageAcceleration (5 points): DO NOT create another project, we are going to add a new file to 'Project2'. From the NetBeans 'File' menu select '**New File**'. In the dialog box under '**File Type**' select '**Java Main Class**' and click '**Next**'. Create a main class with the name '**AverageAcceleration**', verify it is part of 'Project2' and click the '**Finish**' button. Average acceleration is defined as the change of velocity divided by the time taken to make the change, as shown in the following formula:

$$a = \frac{v_1 - v_0}{t}$$

In the **AverageAcceleration** main method, write a program that prompts the user to enter the starting velocity v_0 in meters/second, the ending velocity v_1 in meters/second, and the time span t in seconds and displays the average acceleration. From the Project Navigator you can run the new program by right clicking on the file name and selecting 'Run File' or if the file is active in the editor window press 'Shift - F6'. Here is a sample run:

```
Enter v0, v1, and t: 5.5 50.9 4.5 Enter
The average acceleration is 10.088888888888889
```

Magic8Ball (20 points): DO NOT create another project, we are going to add a new file to 'Project2'. From the NetBeans 'File' menu select 'New File'. On the dialog box under 'File Type' select 'Java Main Class' and click 'Next'. Create a main class with the name '**Magic8Ball**', verify it is part of 'Project2' and click the 'Finish' button. In the main method write a program that simulates the popular children's decision-making toy called the Magic 8 Ball.

To do this, the program will first have to prompt the user to enter his or her question. The user's response should be "stored" in a String-type variable named **response**. Use the **`nextLine()`** method on your Scanner-type object to obtain the *entire* line of text entered by the user.

Once the program has read the entire line of text entered by the user, the program should generate a random number between 0 and 19, which represents the 20 responses which are possible for a Magic 8 Ball:

- It is certain
- It is decidedly so
- Without a doubt
- Yes definitely
- You may rely on it
- As I see it, yes
- Most likely
- Outlook good
- Yes
- Signs point to yes
- Reply hazy try again
- Ask again later
- Better not tell you now
- Cannot predict now
- Concentrate and ask again
- Don't count on it
- My reply is no
- My sources say no
- Outlook not so good
- Very doubtful

The program should generate this random number using the **`Math.random()`** method and should “store” the number in an int-type variable named **randomNumber**.

The program should then use a selection (either nested if or switch structures) to “decide” which Magic 8-ball response to display. Here is what your program’s output must look like:

```
What is your question?  Will it snow tomorrow?   Enter
Your question was: Will it snow tomorrow?
My answer is: Outlook not so good.
```

When you have completed all of the exercises, ZIP the entire project folder and upload this ZIP file to Canvas. In NetBeans, click ‘File’, select ‘Export Project’, click ‘To ZIP…’, select the location where you want the ZIP file to be saved, then click the ‘Export’ button.

If the IDE you are using does not have an export-project-to-zip option, you may have to manually ZIP the file by navigating to the project folder in a file browser and selecting your OS’s “compress folder” option. Under Mac OS, the file browser is called Finder which offers a Compress Folder option. Under Windows, the file browser is called Windows Explorer and it offers a Send to => Compressed (zipped folder) option.