FACULTY OF BUISNESS AND INFORMATION SCIENCE

ACADEMIC SESSION: JAN-APRIL 2017

CC 108: JAVA PROGRAMMING 1

COURSEWORK DEADLINE: **20 MAR 2017**

INSTRUCTIONS TO CANDIDATES

1. This assignment will contribute **30%** to your final grade.
2. This coursework is an individual assignment.

IMPORTANT

Plagiarism is not allowed and any student found to commit such an act, 0 mark will be awarded.

Courseworks must be submitted on their due dates. If a coursework is submitted after its due date, the following penalty will be imposed:

- **ONE day late : 5 marks deducted from the total marks awarded.**
- **TWO days late : 10 marks deducted from the total marks awarded.**
- **THREE or more days late: Assignment will not be marked and 0% will be awarded.**

***For example**: A student scores 60 marks for an assignment that has a total mark of 100.*
If the assignment is submitted one day late, the marks awarded will be 55 marks.

Instructions:

You are required to answer all the questions in this coursework and meets the following requirements.

- Demonstrate all your work with **PRINTSCREEN** photos and include into your report.
- **PRINTSCREEN** all your **CODES** (using IDE) and paste under the **QUESTION**.
- **PRINTSCREEN** all your **OUTPUTS** (using IDE) and paste under the **CODES.**
- A special section after **OUTPUTS** to **EXPLAIN** how you script the codes and how it works.
- Formal Presentation will be conducted (5 marks).

Your report should be comprehensive and must answer all questions with font formats of Arial 12, Normal, 1.5 spacing.

Submit both soft copy and hardcopy of your assignment on mentioned date.

Questions:

1. Code a small program to displays the area and perimeter of a circle with radius **5.5** using the following formula:

   ***perimeter* = 2 * *radius* * 3.14159**

   ***area* = *radius* * *radius* * 3.14159**

2. Script a program that displays the area and perimeter of a rectangle with the width of **4.5** and height of **7.9** using the following formula:

   area* = *width* * *height

3. Script a program that prompts user to enter the number of minutes (e.g., 1,000,000), then output the number of years and days for the minutes. For simplicity, assume a year has **365** days.

   Example output:

```
Enter the number of minutes: 1000000000 Enter
1000000000 minutes is approximately 1902 years and 214 days
```

4. Assume that you know the balance and the annual percentage of interest rate, you want to calculate the interest on the next monthly payment using the following formula:

   interest = balance * (annualInterestRate / 1200)

   Script a program that able to read the balance and the annual percentage interest rate, then displays the interest for coming months. The output is as follow:

```
Enter balance and interest rate (e.g., 3 for 3%): 1000 3.5 Enter
The interest is 2.91667
```

5. Script a program that sorts three integers in decreasing order. The program will prompt user for 3 integers and it will be the input and stored in variables as **a**, **b**, and **c**, respectively. The program sorts the numbers so that ***a* > *b* > *c***.

```
run:
Enter three intergers: 100
200
300
The integers in decreasing order is 300, 200, 100.
```

6. Script a program that plays the popular scissor-rock- paper game. Your program randomly generates a number **0**, **1**, or **2** which will represent scissor, rock, and paper. The program prompts the user to enter a number **0**, **1**, or **2** and displays a message indicating whether the user or the computer wins, loses, or draws. The sample output is as follow:

```
run:
scissor (0), rock (1), paper (2): 1
The computer is scissor. You are rock. You won
```

7. Script a program which convert the value of miles to kilometres. Your program will then display the following table indicating the conversion (note that 1 mile is 1.609 kilometres):

```
run:
Miles                Kilometers
------------------------------
1                    1.609
2                    3.218
3                    4.827
4                    6.436
5                    8.045
6                    9.654
7                    11.263
8                    12.872
9                    14.481
10                   16.09
```

8. Script a program which prompts the user to key-in the number of students. Then followed by each student's name and score. Finally, your program will display the name of the student with the highest score. The outputs are as follow:

```
run:
Enter the number of students: 3
Enter a student name: Apple
Enter a student score: 90
Enter a student name: Pen
Enter a student score: 80
Enter a student name: Pineapple
Enter a student score: 70
Top student Apple's score is 90.0
```

9. Using java, script a method with the function of following header to display an integer in reverse order:

public static void reverse(**int** number)

For example, **1234** displays **4321.**

Script a program that first prompts the user to enter an integer, then your program should be able to displays its reversal result.

```
run:
Enter an integer: 1234567890
0987654321
```

10. Script a class with java which convert Celsius to Fahrenheit and vice versa. Your program should contain the following two methods:

/** Convert from Celsius to Fahrenheit */
public static double celsiusToFahrenheit(**double** celsius)

/** Convert from Fahrenheit to Celsius */
public static double fahrenheitToCelsius(**double** fahrenheit)

The formula for the conversion is:

fahrenheit = (9.0 / 5) * celsius + 32
celsius = (5.0 / 9) * (fahrenheit – 32)

Script a program that call these 2 methods. Then your program should display the table showing the conversion. Sample output is as follow:

```
run:
Celsius         Fahrenheit      |       Fahrenheit      Celsius
=============================================================
10.0            50.0            |       90.0            32.22222222222222
9.0             48.2            |       80.0            26.666666666666668
8.0             46.4            |       70.0            21.11111111111111
7.0             44.6            |       60.0            15.555555555555557
6.0             42.8            |       50.0            10.0
5.0             41.0            |       40.0            4.444444444444445
4.0             39.2            |       30.0            -1.1111111111111112
3.0             37.4            |       20.0            -6.666666666666667
2.0             35.6            |       10.0            -12.222222222222223
1.0             33.8            |       0.0             -17.77777777777778
```

11. Script a program that reads several scores. Then it determines how many scores are above or equal to the average. Then your program should show how many scores are below the average. To stop the program from recording more than your desired scores, enter a negative number to signify the end of the input. Assume that the maximum number of scores is 100. Example output are as follow:

```
run:
Enter a new score: 90
Enter a new score: 80
Enter a new score: 70
Enter a new score: 60
Enter a new score: 50
Enter a new score: -1
count is 5
Average is 70.0
Number of scores above or equal to the average 3
Number of scores below the average 2
```

12. Script a java program that reads student scores. From the score user inputs it should be able to assigns grades based on the following scheme:

Grade is A if score is >=best – 10;

Grade is B if score is >= best - 20;

Grade is C if score is >= best - 30;

Grade is D if score is >= best - 40;

Grade is F otherwise.

The program should prompt the user to enter the total number of students, then prompts the user to enter all the scores, and concludes by displaying the grades. Sample output as follows:

```
run:
Enter number of students: 6
Enter 6 scores: 79 69 59 49 39 29
Student 0 score is 79 and grade is A
Student 1 score is 69 and grade is A
Student 2 score is 59 and grade is B
Student 3 score is 49 and grade is C
Student 4 score is 39 and grade is D
Student 5 score is 29 and grade is F
```

13. Script a method using java which sums all the numbers in the major diagonal in an *n* * *n* matrix of integers using the following header:

public static double sumMajorDiagonal(**double**[][] m)

Write a test program that reads a 4-by-4 matrix and displays the sum of all its elements on the major diagonal.

In linear algebra, the **main diagonal** (sometimes **principal diagonal**, **primary diagonal**, **leading diagonal**, or **major diagonal**) of a matrix {\displaystyle A} is the collection of entries {\displaystyle A_{i,j}} where {\displaystyle i=j}. All off-diagonal elements are zero. The following three matrices have their main diagonals indicated by red 1's:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

The sample output is as follow:

```
run:
Enter a 4 by 4 matrix row by row:
1 2 3 4
1 2 3 4
1 2 3 4
1 2 3 4
Sum of the elements in the major diagonal is 10.0
```

14. Write a program that randomly fills in 0s and 1s into a 4-by-4 matrix, prints the matrix, and finds the first row and column with the most 1s. Here is a sample run of the program:

```
run:
0110
1101
1000
1001
The largest row index: 1
The largest column index: 0
```

15. For a better security, websites with online system set certain rules for user to select passwords when they register. Script a method that checks and validate whether a string input by user is a valid password. Suppose the password rules are as follows:

i. A password must have at least eight characters.

ii. A password consists of only letters and digits.

iii. A password must contain at least two digits.

Script a program that prompts the user to enter a password and displays **Valid Password** if the rules are followed or **Invalid Password** otherwise. The sample output is as follow:

```
run:
Note:
-Password must have at least 8 characters.
-Password must consist only letters and digits.
-Password must contains atleast 2 digits.

Please enter a Password: ABCDE12345
Valid Password
```

16. Write a method that counts the number of letters in a string using the following header:

public static int countLetters(String s)

Write a test program that prompts the user to enter a string and displays the number of letters in the string. The sample output is as follow:

```
run:
Enter a string: Apple Pen Pineapple Pen
The number of letters is 20
```

17. Script a program that creates a **Random** object with seed **1000** and displays the first 50 random integers between **0** and **100** using the **nextInt(100)** method. The sample output are as follow:

```
run:
87 35 76 24 92 49 41 45 64 50 79 59 72 83 36 75 46 2 23 41 22 71 89 2 93 42 49 42 35 76 32 0 52 95 87 31 99 18 79 2 91 5 55 84 71 95 58 87 77 38
BUILD SUCCESSFUL (total time: 0 seconds)
```

------ **END OF QUESTION** ------