King Fahd University of Petroleum and Minerals

###### Information and Computer Science Department

**ICS 102: Introduction to Computer Programming**

**Second Semester 2017-2018 (Term 172)  
Homework No. 1  
[Posted: Tuesday 13th February 2018]  
[Due Date: Tuesday 27th February 2018 @ 11:59 PM (Before Midnight)]**

**Submission Guidelines:**

Submit a zipped file containing the following files:

* Q1.java (Java source file) containing your answer to the programming question no. 1.
* Q2.java (Java source file) containing your answer to the programming question no. 2.
* Q3.java (Java source file) containing your answer to the programming question no. 3.

PLEASE DO NOT INCLUDE .class FILES IN YOUR SUBMISSION

The zipped file should be named as follows:

**HW\_1\_XXXXXXXXX\_YourFamilyName\_Lecture\_Section\_No.zip**

where:

XXXXXXXXX is your 9 digit KFUPM ID.

YourFamilyName is your family name

Lecture\_Section\_No is the number of your ICS 102 lecture section

Submission should be made through your ICS 102 Lecture section Blackboard course page under **HW\_1 Assignment** submission link.

**Important Notes:**

* **Cheating is taken seriously**. Any cheating attempt will result in an F grade in the course.
* **EACH STUDENT IS REQUIRED TO DO THE HOMEWORK ALONE**. COPYING FROM ANY SOURCE IS REGARDED AS CHEATING.
* **No late submissions are allowed**.
* **Submissions via email are not accepted and will be simply ignored**.
* Submission of the homework solution should be in a zipped filed with the format specified above. **Any different formatting/naming will result in reducing the total homework score by half!**
* **You must use proper indentation and meaningful variable names in your programs.**

**Question 01**: Write an interactive **Java** program, containing the **main** function only, which prompts for and reads an integer **n** in the interval [0 . . . 10]. You program must display an error message and terminate if **n** is not in the given interval. If **n** is valid, your program must prompt for and read two floating-point numbers of type **double** and test whether they are equal up to **n** decimal places.

**Note**:

* Your program must handle **InputMismatchException** by displaying an appropriate error message and then terminate.
* Your program must be general and it must behave as in the sample program runs below.

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

**Question 02:** Given a Gregorian date consisting of integers **day**, **month**, and **year**, the following formula, in which all divisions are integer divisions, is used to find the name of the day of the week {\displaystyle w=\left(d+\lfloor 2.6m-0.2\rfloor +y+\left\lfloor {\frac {y}{4}}\right\rfloor +\left\lfloor {\frac {c}{4}}\right\rfloor -2c\right){\bmod {7}},}for the date:

% 7

Where:

* **w** is the day of the week (0 for Saturday, 1 for Sunday, 2 for Monday, 3 for Tuesday, 4 for Wednesday, 5 for Thursday, and 6 for Friday).
* **day** is the day of the month.
* **computedMonth** is the **month** computed as: 3 for March, 4 for April, . . ., 12 for December, 13 for January, and 14 for February.
* **computedYear = year - 1** if the **month** is 1 or 2; otherwise **computedYear = year**
* **k = computedYear / 100**
* **y = computedYear % 100**

Write an interactive Java program that prompts for and reads an integer **year** in the interval [1582 . . . 4000], if the year is not valid, the program displays an appropriate error message and terminates; otherwise, it prompts for and reads an integer **month**. If the month is invalid, the program displays an appropriate error message and terminates; otherwise it prompts for a valid day number for the month. If the entered day number is invalid, the program displays an appropriate error message and terminates; otherwise it computes and displays the day of the week for the date.

**Hint:**

* A Gregorian year is a leap year (a year in which the month of February has 29 days) if it is divisible by 400 or if it is divisible by 4 but not divisible by 100
* You may find the following English poem useful:

Thirty days has September,  
April, June, and November;  
All the rest have thirty-one,  
Excepting February alone,  
And that has twenty-eight days clear  
And twenty-nine in each leap year.

Note:

* Your program must handle **InputMismatchException** by displaying an appropriate error message.
* Your program must not use Java Calendar or Date APIs.
* Your program must not use arrays.
* Your program must contain the **main** method only; it must not call other methods.
* The behaviour of your program must be similar to the sample program runs below:

Sample program runs:

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

**Question 03**: Write an interactive Java program that prompts for and reads a word **word1** and a character **ch1**. If **ch1** does not appear in **word1** an error message is displayed and the program terminates; otherwise the program prompts for and reads another word **word2** and another character **ch2**. If **ch2** does not appear in **word2** an error message is displayed and the program terminates; otherwise the program writes, to the monitor:

* The contents of a new string in which each occurrence of **ch1** in **word1** is replaced by **ch2**.
* The contents of a new string in which each occurrence of **ch2** in **word2** is replaced by **ch1**.

**Note:** The behaviour of your program must be similar to the sample program runs below:

Sample program runs:

|  |
| --- |
|  |
|  |
|  |