Homework 2

Doomsday Calculator

1 Introduction

This assignment will cover basic console I/O, conditional operations, and simple algorithms

2 Problem Description

You have probably seen a movie or watched a show where a character (usually a ”math genius”) asks someone for their birthday, then instantly rattles off the day of the week they were born. While this may seem to require the mind of a savant, the answer can be found using a simple algorithm known as the Doomsday algorithm. Instead of doing this computation in your head, you’re going to implement a Doomsday calculator that can compute the day for any date in the 1990s.

3 Solution Description

The basis of the Doomsday algorithm is simple. The day each year starts on is cyclical such that if the 1st of a given year is on a Monday then the 1st of the next year, 365 days later, will be on a Tuesday. To see this, consider how many weeks there are in those 365 days. $\frac{365}{7} = 52\frac{1}{7}$ = 52 weeks and one additional day. It’s because of the remaining day that each year begins on successive days of the week.

Using the following information, the following algorithm finds a year’s doomsday:

1. Divide the last two digits of the year by 12
2. Modulo the last two digits of the year by 12
3. Divide the answer to Step 2 by 4
4. Sum steps 1-3 and add the century’s ”anchor day”. In the case of the 1900’s for this assignment this value is 3.
5. Modulo the result from the previous step by 7 to get the ”doomsday” for that year. A 0 corresponds to Sunday, 1 to Monday, etc.

The doomsday for a year is a day of the week each of the following dates occur

- January 3rd (or 4th, for a leap year)
- February 28th (or 29th, for a leap year)
- March 7th
- April 4th

- May 9th
- June 6th
- July 11th
- August 8th
- September 5th
- October 10th
- November 7th
- December 12th

Finally, you take the difference between the date you are searching for and the doomsday date of the month (listed above) and add the day you calculated in Steps 5. Modulo that final answer by 7 to get the day of the week for the date you are searching for. Here are some examples:

- February 5th, 1994 = Saturday
- July 1st, 1992 = Wednesday
- October 9th, 1920 = Saturday

After the first date calculation, your program should ask if the user would like to enter another date by either typing `y` or `n`. If the user enters `y` your program should prompt the user for another date to process. If the user enters `n` your program should terminate. Your program will only be tested with positive integers, `y`, or `n`, but you must check for valid dates and leap years. If a valid date is not provided, you should notify the user that the date is invalid and ask for another date.

4 Example Output

The program output should prompt the user for the year, month, and day before returning with the weekday of the entered date

```
$ java Doomsday
Welcome to the Doomsday Calculator!
Enter year (1900-1999): 1993
Enter month (1-12): 3
Enter day: 29
3/29/1993 was on a Monday.

Do you want to enter another date (type ’y’ or ’n’)
y

Enter year (1900-1999): 1978
Enter month (1-12): 1
Enter day: 34
The entered date is invalid!

Do you want to enter another date (type ’y’ or ’n’)
n
$
```

5 Helpful Hints

How to determine if a year is leap year:

1. If the year is evenly divisible by 4, go to step 2. Otherwise, go to step 5.
2. If the year is evenly divisible by 100, go to step 3. Otherwise, go to step 4.
3. If the year is evenly divisible by 400, go to step 4. Otherwise, go to step 5.
4. The year is a leap year (it has 366 days).
5. The year is not a leap year (it has 365 days).

6 Checkstyle

You must run checkstyle on your submission. The checkstyle cap for this assignment is **10** points.
Review the Style Guide and download the Checkstyle jar. Run Checkstyle on your code like so:

```
$ java -jar checkstyle-6.2.2.jar *.java
Audit done. Errors (potential points off):
0
```

The message above means there were no Checkstyle errors. If you had any errors, they would show up above this message, and the number at the end would be the points we would take off.

The Java source files we provide contain no Checkstyle errors. For this assignment, there will be a maximum of **10** points lost due to Checkstyle errors (1 point per error). In future homeworks we will be increasing this cap, so get into the habit of fixing these style errors early!

7 Turn-in Procedure

Non-compiling or missing submissions will receive a zero. NO exceptions

Submit `Doomsday.java` to T-Square. Do not submit any compiled bytecode (`.class` files) or the Checkstyle jar file. When you're ready, double-check that you have submitted and not just saved a draft.

Please remember to run your code through Checkstyle!

7.1 Verify the Success of Your Submission to T-Square

Practice safe submission! Verify that your HW files were truly submitted correctly, the upload was successful, and that the files compile and run. It is solely your responsibility to turn in your homework and practice this safe submission safeguard.

1. After uploading the files to T-Square you should receive an email from T-Square listing the names of the files that were uploaded and received. If you do not get the confirmation email almost immediately, something is wrong with your HW submission and/or your email. Even receiving the email does not guarantee that you turned in exactly what you intended.

2. After submitting the files to T-Square, return to the Assignment menu option and this homework. It should show the submitted files.
3. Download copies of your submitted files from the T-Square Assignment page placing them in a new folder.
4. Recompile and test those exact files.
5. This helps guard against a few things.
   (a) It helps insure that you turn in the correct files.
   (b) It helps you realize if you omit a file or files. [1] (If you do discover that you omitted a file, submit all of your files again, not just the missing one.)
   (c) Helps find last minute causes of files not compiling and/or running.

[1] Missing files will not be given any credit, and non-compiling homework solutions will receive few to zero points. Also recall that late homework will not be accepted regardless of excuse. Treat the due date with respect. The real due date is midnight. Do not wait until the last minute!