

SCIT

School of Computing and Information Technology

Autumn 2017

CSIT111/811 — Programming Fundamentals

Assignment 3 (15 marks)

Due Time and Date:

Due by May 21, 8 pm, Resubmission is open till May 28, 8 pm.

General Requirements:

- You should create your programs with good programming style and form using proper blank spaces, indentation and braces to make your code easy to read and understand;
- You should create identifiers with sensible names;
- You should make comments to describe your code segments where they are necessary for readers to understand what your code intends to achieve.
- Logical structures and statements are properly used for specific purposes.
- Read the assignment specification carefully, and make sure that you follow whatever directed in this assignment. In every assignment that you will submit in this subject, you must put the following information in the header of your assignment:

```
/*-----
My name:
My student number:
My course: CSIT111 or CSIT811
My email address:
Assignment number: 3
-----*/
```

Objectives

This assignment requires you to write a program in Java that calculates the day of week and the week of month for a given date, as well as to print the calendar of the month where the given date exists.

Background

2017 MAY						
SUN	MON	TUE	WED	THU	FRI	SAT
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

www.free-printable-calendar.com

For a given date, we can use the Gregorian calendar to find the corresponding the day of the week, and the week of the month. For example, May 27 2017 is a Saturday and locates in the fourth week of May 2017. In general, we can use the following formula (Zeller's congruence) to calculate the day of a week for any given date after October 15 1582.

$$h = \left(q + \left\lfloor \frac{13(m+1)}{5} \right\rfloor + K + \left\lfloor \frac{K}{4} \right\rfloor + \left\lfloor \frac{J}{4} \right\rfloor + 5J \right) \bmod 7,$$

where

- h is the day of the week (0 = Saturday, 1 = Sunday, 2 = Monday, 3=Tuesday, 4=Wednesday, 5=Thursday, 6 = Friday)
- q is the day of the month
- m is the month (3 = March, 4 = April, 5 = May, 6=June, 7=July, 8=August, 9=September, 10=October, 11=November, 12=December, 13=January, 14 = February)
- K is the year of the century (year mod 100).
- J is the zero-based century (year/100). For example, the zero-based centuries for 1995 and 2000 are 19 and 20 respectively.

For example, For 1 January 2000, the date would be treated as the 13th month of 1999, so the values would be: q=1, m=13, K=99, J=19, so the formula is

$$\begin{aligned} h &= (1 + [13 * (13 + 1) / 5] + 99 + [99 / 4] + [19 / 4] + 5 * 19) \bmod 7 \\ &= (1 + [182 / 5] + 99 + [99 / 4] + [19 / 4] + 95) \bmod 7 \\ &= (1 + [36.4] + 99 + [24.75] + [4.75] + 95) \bmod 7 \\ &= (1 + 36 + 99 + 24 + 4 + 95) \bmod 7 \\ &= 0 \\ &= \text{Saturday} \end{aligned}$$

However, for 1 March 2000, the date is treated as the 3rd month of 2000, so the values become q=1, m=3, K=00, J=20, so the formula is

$$\begin{aligned} h &= (1 + [13 * (3 + 1) / 5] + 00 + [00 / 4] + [20 / 4] + 5 * 20) \bmod 7 \\ &= (1 + [10.4] + 0 + 0 + 5 + 100) \bmod 7 \\ &= (1 + 10 + 5 + 100) \bmod 7 \\ &= 4 \\ &= \text{Wednesday} \end{aligned}$$

Request

You will create one program called `MyCalendar.java`. You should first implement the `MyCalendar` and `MyDate` classes from the following UML class diagrams. You can add extra attributes and methods, but the attributes and methods shown in the UML diagrams can't be modified.

MyCalendar
- myDate: MyDate - day: Day
+ <u>Main(String[] args)</u> + MyCalendar(MyDate myDate) + dayOfWeek(): Day + weekOfMonth(): int + printCalendar(): void

- Day is an enumeration data type which contains the days of the week, i.e., Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday.
- MyCalendar(MyDate myDate) is the construction method of the class MyCalendar.
- dayOfWeek() method will return the day of the week for the given date.
- weekOfMonth() method will return the week of the month for the given date.
- printCalendar() method will print the calendar of the month for the given date.
- Main(String[] args) method will read a given date from the command line, the format of the given date is dd/mm/yyyy. You can assume the date input format is almost correct, but the input date may be invalid. For example, 31/02/2017 is not a valid date. If the input date is not a valid date, the program will ask user to input another valid date through keyboard. This process will be repeated until a valid date is input by the user. The valid input date will be used to create an instance of class MyCalendar. The Main() method will call the dayOfWeek() method, weekOfMonth() method, and printCalendar() method of myCalendar object respectively, and to display the day of the week, the week of the month, and the calendar of the month for the input date by the user.

MyDate
- day: int - month: int - year: int
+ MyDate(int day, int month, int year) + getDay(): int + getMonth(): int + getYear(): int + isValid(): boolean

- MyDate(int day, int month, int year) is the constructor of the class MyDate.
- getDay() method will return the day.
- getMonth() method will return the month.
- getYear() method will return the year.
- isValid() method will return a true when the date is valid, and a false otherwise (notes: January, March, May, July, August, October, December has 31 days. April, June, September, November has 30 days. February has 29 days in a leap year, and 28 days in a common year.).

Example of the program output:

```
java MyCalendar 29/02/2017
```

29/02/2017 is not a valid date, please re-input a valid date: 27/05/2017

27/05/2017 is a Saturday and locates in the fourth week of May 2017

The calendar of May 2017 is:

SUN	MON	TUE	WED	THU	FRI	SAT
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

(Each column indicates a day and the column width are fixed to 3 characters. The distance between two adjacent columns is fixed to three characters as well.)

Submission:

Use `submit` program to submit your assignment:

```
submit -u your_login_name -c CSIT111 -a 3 MyCalendar.java
```

For CSIT811 students, submit your assignment through:

```
submit -u your_login_name -c CSIT811 -a 3 MyCalendar.java
```

- **The submitted file name must be `MyCalendar.java`. Files with other names will not be tested and marked.**
- **Submission via email is NOT acceptable.**

NOTES:

1. Submit your assignment before the due date. You can resubmit later if necessary. Only the latest submission will be tested and marked providing that you have submitted at least one meaningful solution before the due date. Otherwise, resubmitted assignments will not be assessed.
2. Submission via email is not acceptable.
3. Assignments without properly filled assignment headers will not be marked
4. Enquiries about the marks can only be made within a maximum of 1 week after the assignment results are published. After 1 week the marks cannot be changed.

Mark deductions: compilation errors, incorrect result, program is not up to spec, poor comments, poor indentation, meaningless identifiers, required numeric constants are not defined, the program uses approaches which has not been covered in the lectures. The deductions here are merely a guide. Marks may also be deducted for other mistakes and poor practices.