INTRODUCTION TO SCIENTIFIC AND ENGINEERING COMPUTATION
Assignment 1

March 2018

Objective: For a given date, determine which weekday (Monday through Sunday) the date corresponds to, and how many times the given day-and-month falls on the same weekday starting with the next year and ending on a given year.

Description

To identify which weekday a given date corresponds to, we can use Zeller's congruence:

$$\left(q + \left\lfloor \frac{13 \cdot (m+1)}{5} \right\rfloor + k + \left\lfloor \frac{k}{4} \right\rfloor + \left\lfloor \frac{j}{4} \right\rfloor + 5j \right) \bmod 7$$

where:

- q is the day.

- m is the month. But if the month is January or February, we add 12 to the month and subtract 1 from the year.

- j is the zero-based century of the year. In a 4-digit year, this is the first two digits.

- k is the offset of the year within its century. In a 4-digit year, this is the last two digits.

Here are some example values:

- for 2018-03-05: $j = 20, k = 18, m = 3, q = 5$

- for 2018-02-25: $j = 20, k = 17, m = 14, q = 25$

- for 1990-06-13: $j = 19, k = 90, m = 6, q = 13$

The result of the formula determines the day of the week where 0 corresponds to Saturday, 1 to Sunday, ..., and 6 to Friday.

Assignment

Write a C program that will get a date and an end year from the user and print which weekday that date corresponds to, along with the number of the years when the same day-and-month falls on the same weekday, starting from the next year until the user supplied end year.

For example, if the user gives the date as 1900-1-1 and the end year as 1999, the program should report that the date 1900-1-1 is a Monday and how many more dates in the form YYYY-1-1 fall on a Monday, where YYYY are the years between 1901 and 1999 (14 of the January 1's in that year range fall on a Monday).

Your program should also check whether the date given by the user is valid or not.

This is the outline of the program:

1. Get a date from the user.

2. Check whether the date is valid. (Hint: Determine the maximum number of days in the given month. *You can assume that the user will never enter Feb 29.*) If the date is invalid, report it and end the program with a failure code.

3. Get an end year from the user.

4. Determine the day of the week and print it.

5. For the remaining years as described above, count how many of the same day-and-month fall on the same weekday and print the result.

6. End the program with a success code.

Below are some example runs:

```
Enter date [year month day]: 1900 1 1
Enter end year: 1999
It's a Monday.
Same day-and-month on same weekday between 1901 and 1999: 14

Enter date [year month day]: 2018 11 31
Invalid date.
```

Rules

- Your source code file has to have the name "assignment1.c".

- Your program will be compiled using the following command on a Linux system. If it cannot be compiled and linked using this command, it will not be graded (failed submission).

```
gcc -std=c99 -Wall -Werror assignment1.c -o assignment1
```

- Your program will be checked using an automatic checker. Therefore, make sure you print the messages exactly as given in the example runs.

- Do NOT use any construct that hasn't been covered in the course thus far such as defining your own functions. Also use no C++ features such as cout and cin.

- Do NOT use any external functions except for printf and scanf.

- Make sure your coding style is proper and consistent. Use the clang-format tool if necessary. Don't use any variable names in a language other than English.

- This is an individual assignment. Collaboration in any form is NOT allowed. No "working together", no sharing code in any form including showing code to your classmates to give them ideas.

- All the code you submit must be your own. Don't copy/paste any piece of code from any resource including anything you've found on the Internet.