CS19003 : Programming and Data Structures Laboratory
Assignment 2 : Conditional Statements in C
Date: 13-April-2021

Problem Statement:

It has been almost two weeks since Detective Byomkesh Bakshi and his assistant Ajit were being involved to solve a long-standing murder mystery which was rejuvenated by the pair of murders that happened in the city of joy, Kolkata. There must be some clue hidden within, which are yet to be cracked – may be, the dates of murder can give a definitive clue, who knows! Thinking so, Mr. Bakshi was discussing with Ajit about the characteristic of the dates when these murders happen and the last murder date in particular.

However, they found out that, in this eastern part of India, often Bengali calendar is followed (which is a *lunar calendar* as opposed to the solar Gregorian/English calendar that we universally access) for the dates. In Bengali calendar, the names of the months and the number of days within are different than Gregorian/English calendars, which is shown below:

Month No.	Month Name	Days	Month No.	Month Name	Days	Month No.	Month Name	Days
01	Vaisakha	31	**05**	Bhadra	31	**09**	Pausha	30
02	Jyeshtha	31	**06**	Ashwina	31	**10**	Magha	30
03	Ashadha	31	**07**	Kartika	30	**11**	Phalguna	29 / 30*
04	Shraavana	31	**08**	Agrahayana	30	**12**	Chaitra	30

* `Phalguna` has 30 days, if the corresponding year in which it belongs to is a leap-year; otherwise it has 29 days.

Now, Mr. Bakshi asked Ajit to figure out the differences in these two dates of the murder according to the Bangali calendar in various forms. Moreover, Ajit also tried to find out the characteristics of the last murder date as well and accordingly reported to Byomkesh for analyzing the murder cases.

Can you write a C-program to generalize/automate this process of performing date arithmetic and finding date characteristics? In particular, your program will do the following (only using I/O, conditional statements, arithmetic/logical/assignment operators and expressions; *without using loop statements, arrays and functions*):

- Take from user two murder dates, in `DD/MM/YYYY` format. Then, also take the day of the week only for the first (earlier) date, indicated by a number, such as $[0, 1, \ldots, 6] \leftarrow [\texttt{Sunday}, \texttt{Monday}, \ldots, \texttt{Saturday}]$.

- Print the difference between the two murder dates in three formats as follows:
    - In terms of the number of days (total) in between these two dates,
    - In terms of the number of years, number of months and number of days in between these two dates,
    - In terms of the number of elapsed weeks and remaining days in between these two dates.

- Print the status of second (later) murder date with the following information:
    - Given the earlier murder happened in the day of the week as given by the input, determine the day of the week for the later murder date.
    - Mention the name of the month according to the Bengali calendar on which the last murder happened.
    - Estimate whether the year is a leap-year or not when the last murder took place.
    - Present the year of the last murder in English words.

Example Execution Details:

Sample-1:

```
**** User Input of Date Details ****
    ++ Enter First (earlier) Date (DD/MM/YYYY format): 14/06/1424
    ++ Enter Second (later) Date (DD/MM/YYYY format): 14/12/1428
    ++ Enter the Day of the Week for First Date (0-6 for Sun-Sat): 0

**** The Date Difference ****
    ++ In DAYS (only): 1642 days
```

```
    ++ With BREAK-UP: 4 years, 6 months, 0 days
    ++ In WEEK-DAYS: 234 weeks and 4 days

**** The Second Date Status ****
    ++ The Day of the Week: Thursday
    ++ The Name of the Month: Chaitra
    ++ The Year Characteristics: LEAP-YEAR!
    ++ The Year in English: One Thousand Four Hundred Twenty Eight
```

Sample-2:

```
**** User Input of Date Details ****
    ++ Enter First (earlier) Date (DD/MM/YYYY format): 26/05/1119
    ++ Enter Second (later) Date (DD/MM/YYYY format): 18/02/1920
    ++ Enter the Day of the Week for First Date (0-6 for Sun-Sat): 4

**** The Date Difference ****
    ++ In DAYS (only): 292458 days
    ++ With BREAK-UP: 800 years, 8 months, 23 days
    ++ In WEEK-DAYS: 41779 weeks and 5 days

**** The Second Date Status ****
    ++ The Day of the Week: Tuesday
    ++ The Name of the Month: Jyeshtha
    ++ The Year Characteristics: LEAP-YEAR!
    ++ The Year in English: One Thousand Nine Hundred Twenty
```

Sample-3:

```
**** User Input of Date Details ****
    ++ Enter First (earlier) Date (DD/MM/YYYY format): 01/01/0400
    ++ Enter Second (later) Date (DD/MM/YYYY format): 30/07/0900
    ++ Enter the Day of the Week for First Date (0-6 for Sun-Sat): 2

**** The Date Difference ****
    ++ In DAYS (only): 182837 days
    ++ With BREAK-UP: 500 years, 6 months, 29 days
    ++ In WEEK-DAYS: 26119 weeks and 4 days

**** The Second Date Status ****
    ++ The Day of the Week: Saturday
    ++ The Name of the Month: Kartika
    ++ The Year Characteristics: NOT LEAP-YEAR!
    ++ The Year in English: Nine Hundred
```

Sample-4:

```
**** User Input of Date Details ****
    ++ Enter First (earlier) Date (DD/MM/YYYY format): 29/11/2019
    ++ Enter Second (later) Date (DD/MM/YYYY format): 01/12/2019
    ++ Enter the Day of the Week for First Date (0-6 for Sun-Sat): 6

**** The Date Difference ****
    ++ In DAYS (only): 1 days
    ++ With BREAK-UP: 0 years, 0 months, 1 days
    ++ In WEEK-DAYS: 0 weeks and 1 days

**** The Second Date Status ****
    ++ The Day of the Week: Sunday
    ++ The Name of the Month: Chaitra
    ++ The Year Characteristics: NOT LEAP-YEAR!
    ++ The Year in English: Two Thousand Nineteen
```

Submit a single C source file. Do not use *loop-statements, arrays, functions* and global/static variables.