Project 1: Birthdays (Due Sept. 23 at midnight)

In folklore, the day of the week you are born on says a lot about your personality. *Monday's Child* is a nursery rhyme that goes:

> Monday's child is fair of face,
> Tuesday's child is full of grace,
> Wednesday's child is full of woe,
> Thursday's child has far to go,
> Friday's child is loving and giving,
> Saturday's child works hard for a living,
> But the child who is born on the Sabbath day
> Is fair and wise and good in every way.

In this project, you will write a program that asks the user for their birthday, then determine on what day they were born.

Project Description

Getting straight to the point, here's an example run:

```
Enter your birthday in YYYY-MM-DD format: 1887-04-20
You were born on a Wednesday!
```

As part of your project, you will write five functions to perform various parts of the calculation. They are listed below, with details in the starter file:

- `is_leap_year(year)`
- `is_gregorian_date(year, month, date)`
- `is_valid_date(year, month, date)`
- `weekday_of(year, month, date)`
- `weekday_name(weekday)`

For all of the above functions, you may assume that the arguments will be integers.

Finally, to tie all the functions together, you will write a function `main()` that contains the code that gets user input and prints out the appropriate response. Refer to the lecture slides for how to make it so the function is only called when the file is run, but not when it is used as a module.

Final note: Python has a `datetime` module, which **you are not allowed to use for this project.** The only module you should need for this project is `math`, which provides the `floor` function.

Zeller's Congruence

The `weekday_of()` function implements Zeller's congruence:

$$\left(D + \left\lfloor \frac{13(M+1)}{5} \right\rfloor + Y + \left\lfloor \frac{Y}{4} \right\rfloor + \left\lfloor \frac{C}{4} \right\rfloor + 5C \right) mod\, 7$$

In this formula:

- The square brackets with a bottom border is the floor function (ie. rounding down to the nearest integer).
- *D* is the date of the month
- *M* is the month (but see the adjustments below)
- *Y* is the year *without* the century (but see the adjustments below)
- *C* is the century (but see the adjustments below)

One caveat for Zeller's formula is that it considers *February* to be the last month of the year, so January and February are months 13 and 14 of the previous year. This adjustment is done before the century and year is calculated. For example, if the call to this function was `weekday_of(2000, 2, 3)` (ie. February 3, 2000), the variables in the equations would be:

- *D* would be 3
- *M* would be 14
- *Y* would be 99
- *C* would be 19

Peer Evaluation

Once you are done, you need to fill out the peer evaluation for this project. Please do this even if you worked alone, since it will give me feedback about how to improve the project in the future.