Assignment 3

due Thursday 24 October

Complete the code for the functions specified below. Place your work in a single textfile named `a3.py`. Submit this using Canvas on or before the due date indicated above.

Write a Python function `cal(month, year)` that takes two integer arguments representing a month and a year and that prints out a calendar page for that month in the following format (for October 2019). The program need only cope with months in years 2000 to 2099.

```
Su Mo Tu We Th Fr Sa
       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
```

You must also write the two functions listed below:

num_days(month, days) return the number of days in the specified month

start_day(month, year) return the day of the week on which the first of the month falls (with 0 for Sunday, 1 for Monday and so on).

Notes

1. Do not use any of Python's date manipulating capabilities such as `datetime` for this.

2. To determine the number of the days in a given month, remember the ditty "Thirty days hath September . . ." (look it up if you have forgotten). Note also that the leap years between 2000 to 2099 are years divisible by four.

3. There are various ways to figure out the day of the week on which a particular date falls. The following (inefficient but adequate) approach is to be used here. The first of Janurary 2000 fell on a Saturday. Step forward from that date one day at a time $(1/1/2000, 2/1/2000, 3/1/2000, \cdots)$ keeping track of the day of the week as you go until you reach the date in question. It may be helpful to develop a helper function `next_day(d, m, y)` that returns (as a triple of ints) the date immediately following $d/m/y$.

4. To print an integer so that it occupies precisely three characters width in the output use Python's output formatting feature:

   ```
   print("%2d" % x)
   ```