

CS 121 – Lab #10 – Calendar Functions

The purpose of today's lab is to practice writing and calling functions. The specific application deals with calendar-related functions. On your USB drive or account, create a new folder called lab10 and do all of today's work in there. On the class Web site, download the file calendar.py. You will make changes to this file to complete its implementation.

Part 1 – Calendar function implementations

At the top of calendar.py, we need implementations of these functions:

- isLeapYear() – given a year number, is it a leap year or not
- findMonthLength() – determine the number of days in a month, given the month and year
- findJulianDay() – calculate the Julian day of a certain day/month/year
- findZodiac() – determine the Zodiacal sign when given the month and day

A fifth function, findHoroscope(), has already been implemented for you. Currently, the implementations of the first 4 functions are just skeletons just to make the compiler happy. You need to complete the implementations of these 4 functions. At the bottom of calendar.py, the main program contains code that will help you test your implementations. Please do not change these function calls. For the findZodiac() method, use the following definitions of the signs:

Sign	Date range
Aries	March 21 – April 19
Taurus	April 20 – May 20
Gemini	May 21 – June 20
Cancer	June 21 – July 22
Leo	July 23 – August 22
Virgo	August 23 – September 22
Libra	September 23 – October 22
Scorpio	October 23 – November 21
Sagittarius	November 22 – December 21
Capricorn	December 22 – January 19
Aquarius	January 20 – February 18
Pisces	February 19 – March 20

Here is a hint for implementing findZodiac() – you can convert the month/day into a single value equal to $100 * \text{month} + \text{day}$. Then this single number can be compared to a range of values. For example, Libra would correspond to the numbers 923 to 1022, inclusive. You need to be careful with the definition of Capricorn since it wraps around to the next year.

Run calendar.py. Examine the output to verify that your function implementations are correct.

Part 2 – Interactive I/O in the main program

The main program is located at the end of the source file `calendar.py`. After testing the various calendar functions, the program asks the user to enter a date. It then calls the various calendar functions for that date.

To finish the main program, you need to correct the five assignment statements for the variables `leapYear`, `daysInMonth`, `julianDay`, `sign` and `horoscope`. Each of these assignment statements needs to perform a function call.

When you are finished, please demonstrate your completed program to the instructor or lab aide.

Have fun!