**Name: Session:**

**Programming I**

**Lab Exercise 10.31.2023**

**Some Problems to Solve**

Write a program to solve the following problems. You may use any technique you have previously used. Make sure you document your source code as follows:

#Author: Jane Smyth

#Lab Exercise 11.2.2022 Problem 1

#Program Description

When your programs are working, submit the documented source code.

1. Using a for loop, write a program that prints out the decimal equivalents of 1/2, 1/3, 1/4,..., 1/20.
2. Write a function that uses a loop that calculates integer exponentials. Your program should ask the user for a *base* and an exponent *exp* as integers and calculates *baseexp*. **Do not use any math library functions**. Example: 25 should print out 32.
3. Write a function convertBase that is provided a base 10 number and a base to convert to. The function should then return a number representing the base. For example 65 base 10 should return 1000001 in base 2. Hint: to convert to any base perform repeated integer division by the base to convert to and save the remainders. Your converted number is the reverse of the saved remainders
4. Zeller’s algorithm computes the day of the week on which a given date will fall (or fell). In this exercise, you will write a program to run Zeller’s algorithm on a specific date. You will need to create a new file for this program, zellers.py. The program should use the algorithm outlined below to compute the day of the week on which the user’s birthday fell in the year you were born and print the result to the screen.

Your program should ask for the month as a number between 1-12 where March is 1 and February is 12. If born in Jan or Feb, enter previous year (see the notes below). In the end, print out the name of the user and on what day of the week they were born.

Use the following format for inputting the date: mm/dd/yyyy

Zeller’s algorithm is defined as follows:

Let A, B, C, D denote integer variables that have the following values:

A = the month of the year, with March having the value 1, April the value 2, . . ., December the value 10, and January and February being counted as months 11 and 12 of the preceding year (in which case, subtract 1 from C)

B = the day of the month (1, 2, 3, . . . , 30, 31)

C = the year of the century (e.g. C = 89 for the year 1989)

D = the century (e.g. D = 19 for the year 1989)

Note: if the month is January or February, then the preceding year is used for computation. This is because there was a period in history when March 1st, not January 1st, was the beginning of the year.

Let W, X, Y, Z, R also denote integer variables. Compute their values in the following order using integer arithmetic:

W = (13\*A -1) / 5

X=C/4

Y=D/4

Z = W + X + Y + B + C -2\*D

R = the remainder when Z is divided by 7

The value of R is the day of the week, where 0 represents Sunday, 1 is Monday, . . ., 6 is Saturday. If the computed value of R is a negative number, add 7 to get a non negative number between 0 and 6.

**To make things easier, I have placed some starter code on the server. You may use this if you wish**.

Run some test cases-try today’s date, your birth date, and whatever else interests you!

This uses the Gregorian calendar, which is our current standard. There are other calendars in use in the world. As the Gregorian calendar was devised in 1582, fixing the cumulative errors of the preceding Julian calendar, it is not clear that Zeller's algorithm will work properly for dates before 1582.