IC152: Assignment 4
Loops Continued, Lists, and Problems involving Lists

In this assignment, continue with writing loops, use lists, list methods, and solve some problems involving lists. You have to write one python file for each question. So, in total there will be 4 python files to submit for this assignment.

If you are solving this assignment in the A11's PC Lab: Keep Fn + F9 pressed during the start of your machine (do not repeatedly press, keep it continuously pressed), and then select the second option with "ubuntu". Please check if you are able to log in to moodle, or else change the machine.

Problem 1: Loops Continued.
Write a common python file "q1Fibonacci.py" and prompt the user with "q1 part a input (integer): ", "q1 part b input (list): " and so on for different parts.

a. Define a function that takes the input n and prints the Fibonacci series for 'n' terms. Implement using lists/storage. The input n is given by the user.
   About Fibonacci series:

$$F_0 = 0, F_1 = 1$$

$$F_n = F_{n-1} + F_{n-2}$$

For $n > 1$

$\Rightarrow$ Fibonacci Series:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ...

b. Define a function to find the majority element in an array. The function takes a list as a parameter and returns the majority element if it exists. If the majority element does not exist then return -1.
   The majority element is that element that occurs more than (n/2) number of times where n is the size of the array.
   For eg: [1,2,3,1,1,2,1,1] : ans = 1
   You will need to create a list using string input given by the user. Try different corner cases for user inputs like empty list ([ ]), list with negative integers, etc.

c. Write a program to print the first number which repeats itself in the given array. If there is no such element print “no repeating element exists”
   For eg: [4,5,2,4,7,5]: print 4

d. Take an input **n** for the user and print the first n rows of the following pattern (including the preceding spaces).

```
    1
   121
  12321
 1234321
123454321
```

e. Write a program to remove the repeated elements from a list after taking input from the user.
   Eg: input: [1,2,1,3,7,6,3]   Output: [1,2,3,7,6]

Problem 2: List Operations.

Write a common python file "q2ListOperations.py" and prompt the user with "q2 part a input (list): ", "q2 part b input (integer): " and so on for different parts.

a. Write a program that prints the input list in ascending order and descending order using list.sort() method of lists.
b. Replicate a given list x number of times where x is an integer from user using '*' operator.
   Eg. input_list = [1, 5, 3]
   Output_list = [1, 5, 3, 1, 5, 3, 1, 5, 3]
c. Write a program to print the maximum and minimum number in the list using max() and min() functions.
d. Find the position of an element in the list using list.index() function. Input the element k from the user.
   Eg: input: [10, 30, 20],   k = 30
   Output: 1
   Output should be -1 in case an element is not in the list.
e. Print the sum of all elements in the list using the sum(list_name) function.

Problem 3: Polynomial functions using lists.

Write a common python file "q3Polynomial.py" and prompt the user with "q3 part a input (list): ", "q3 part b input (integer): " and so on for different parts.

a. Take the coefficients of the polynomial function from the user as a list. Now define a function to return the coefficient of the given degree provided by the user = n;

Example: input: [2, 5, 0, 1] , n = 2

Output: 5

b. Given the polynomial function: f(x) = $4x^3 - 6x^2 + 0x - 1$
Write a program that takes the input of x from the user and returns the value of the polynomial for the given value of x.

c. Take input of two polynomial functions from the user in the form of lists as above. Now Write a program to evaluate the sum of the two polynomial functions and print the final polynomial.
Example input: f1(x) = $2x^3 + 3x^2 - 10$;

f2(x) = $4x^5 + 1x^3 + 2x + 1$

Output: “f(x) = $4x^5 + 3x^3 + 3x^2 + 2x - 9$”

Problem 4: 2D lists.

Write a common python file “q4_2Dlists.py” and prompt the user with “q4 part a input (string): ”, “q4 part b input (integer): ” and so on for different parts.

a. Write a program to take the input of a 2D matrix from the user. Start with taking input for a number of rows and a number of columns in the matrix. Then input the matrix row by row.
Print this matrix in a matrix format.
Now take an input k from the user and print whether this element exists in the matrix or not. If the element does exist, print the [row, col] where the element was found, otherwise print “element not found in the matrix”.
If there are multiple instances of k, print all the indexes.

b. Write a function that takes a square matrix of size n as input and prints whether the input matrix is
   i. Symmetric or not
   ii. Skew-Symmetric or not

Note:

Symmetric matrix: A matrix is symmetric iff (if and only if) it is equal to its transpose. All entries above the main diagonal of a symmetric matrix are reflected into equal entries below the diagonal.
Skew symmetric matrix: A matrix can be skew-symmetric only if it is a square matrix. If the transpose of a matrix is equal to the negative of itself, the matrix is said to be skew-symmetric. This means that for a matrix to be skew-symmetric, the diagonal elements of the matrix are 0.

Extra/Bonus question 1: Write a program to find roots of a given polynomial.
Extra/Bonus question 2: Write a python program to sort a list without using any inbuilt function (like list.sort()) . You can use any list or create your own.

Create the folder having your python files, with name having your roll number followed by "_assignment4" (don't use inverted commas in folder name), compress the folder with .zip extension and submit it on moodle.

Make sure that you delete all your files from the lab PC/Laptop, and shut it down before you leave.