The objective of this lab is to:

1. Practice 2-D dynamic arrays and double pointers.
2. Practice string functions, structures, user-defined data types and array of objects.

Instructions!

1. Please follow dress code before coming to the lab. Keep your student identity cards with you.
2. This is an individual lab, you are strictly **NOT** allowed to discuss your solutions with your fellow colleagues, even not allowed asking how is he/she is doing, it may result in negative marking. You can **ONLY** discuss with your TAs or with me.
3. Strictly follow good coding conventions (commenting, meaningful variable and functions names, properly indented and modular code.
4. Save your work frequently. Make a habit of pressing **CTRL+S** after every line of code you write.
5. Beware of **memory leaks** and **dangling pointers**.

Task 01: [5 Marks]

1. Write a program that uses a structure named *MovieData* to store the following information about a movie:
   Title
   Director
   Year Released
   Running Time (in minutes)

   The program should create two *MovieData* variables, store values in their members, and pass each one, in turn, to a function that displays the information about the movie in a clearly formatted manner.

Task 02: [10 Marks]

Write a program that stores a class of students for Object Oriented Programming course where the class has 30 students in it. Create a structure that stores following data about a student:

Students's Name
Students's RollNumber
CGPA

To store all students of the class, the program should keep an array of size 30. Each element of the array should store a different student in the class. Write a function that populates the array of structures by reading data from the file named "Students.txt". The file Students.txt is provided as part of this lab which contains data of 30 students. The name, roll number and CGPA is comma separated. While every student's data starts on a new line.

Your program should display a list of students with students's Roll Number, names and CGPA.

Implement the following functions:

1. ***populateArray*** – to populate the array of student by reading the data from Student.txt file.
2. ***FindCR*** – this function should return the student with highest CGPA in the class.
3. ***printDropOutStudents*** – this function should print a list of students (Name, Roll Number, CGPA) whose CGPA is less than 2.70.
4. ***printProbations*** – this function should display a list of students (Name, Roll Number, CGPA) whose CPGA is less than 2.3 but greater than 2.7.
5. ***printStudentData*** – this function should accept a student's roll number as input and print its name, CGPA, degree, session, section and department. You can find the degree, session, section and department from the roll number.

   For example: If the function gets **BSEF18M021** as input the function should print following information:

   Name: Areeba Imtiaz
   Degree: BS Software Engineering

Session: 2018-2022
Section: Morning
Department: Department of Software Engineering, Old Campus.

Call each of these functions in menu based *main( )* program to demonstrate your work. Use appropriate prompts for input and output to make your program easy to use. Furthermore, do input validation where required.

Task 03: [5 Marks]

Write and program that should create a square matrix of user-defined size, populate the matrix with user given values, and determine whether the matrix is a diagonal matrix or not. Where, a diagonal matrix is a matrix in which the entries outside the main diagonal are all zero. The diagonal entries themselves may or may not be zero.

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & -2 \end{bmatrix}$$