National University of Computer and Emerging Sciences, Lahore Campus

Course:	**Object Oriented Programming Lab**	**Course Code:**	**CL217**
Program:	**BS (Computer Science)**	**Semester:**	**Fall 2020**
Duration:	**2 Hours**	**Total Marks:**	**100**
Paper Date:	**4-Nov-2020**	**Weight**	**30%**
Section:	**All**	**Page(s):**	**2**
Exam:	**Lab Midterm**	**Roll. No**	

Read below Instructions Carefully:

- Understanding the question statement is also part of the exam, so do not ask for any clarification. In case of any ambiguity, make suitable assumptions.
- You have to complete exam in 2 hrs. An extra 10 minutes are for submission.
- For Q1, create a folder named by your Roll number in the format 19L-9085 which should contain Matrix.cpp, Matrix.h and Driver.cpp. Submission Path is \\cactus\Xeon\Fall 2020\Hamna Waseem\OOP Labs\Midterm\BCS-3X\Q1. (X is your section name.)
- For Q2, submit a single file (containing function and main) named as 19L-9085.cpp on \\cactus\Xeon\Fall 2020\Hamna Waseem\OOP Labs\Midterm\BCS-3X\Q2. (X is your section name.)
- Submit both questions in a zipped folder (named as your roll number) on Slate in assignment titled as **OOP- Lab Midterm Submission**.
- Your code should be intended and commented properly. Use meaningful variable names.
- It is your responsibility to save your code from being copied. All matching codes will be considered cheating cases. PLAGIARISM will result in forwarding of case to Disciplinary Committee and negative marks in Midterm.

Question No. 01: Marks: 60

Any matrix can be broken down to a chunk of multiple sub-matrices. This process of splitting up the matrix into sub-matrices is called Matrix Partitioning.

For Example the matrix A is split into following 4 matrices.

$$A= \left[\begin{array}{ccccc:cc} 1 & 26 & 8 & 56 & 58 & 95 & 63 \\ \hdashline 3 & 4 & 56 & 89 & 53 & 102 & 36 \end{array}\right]$$

A1 = [1 26 8 56 58] A2 = [95 63]

A3 = [3 4 56 89 53] A4 = [102 36]

Matrix A has 2 rows and 7 columns that can be split horizontally and vertically into 4 sub matrices of different dimensions. In this example, the split is made on **1st row** and **5th column**. Even though there can be multiple splits in a matrix but we’ll stick to four sub matrices for now.

Consider the following class that holds the 2-D matrix along with its dimensions.

```
Class my2D
{
private:
        int ** ptrto2D;
        int rows;
        int cols;
public:
        --------
};
```

Write a function with the following prototype that handles the partitioning of matrix:

my2D * SplitUp (const my2D & matrix);

The function returns the pointers pointing to the 4 partitioned matrices. The row and column from where to split will be taken as input from user in the function.

<u>A function is provided in function.txt file placed at \\cactus\Xeon\Fall 2020\Hamna Waseem\OOP Labs\Midterm that you might need for this questions' proper working.</u>

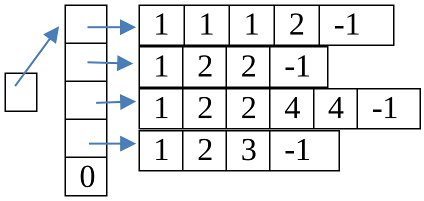
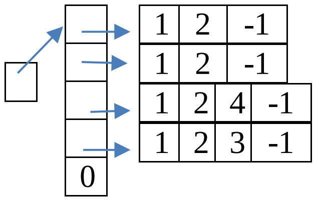
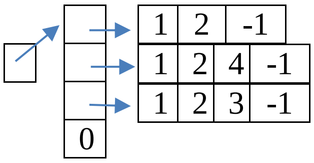
Your main program should do the following:

- Take input of matrix dimensions.
- Populate the matrix.
- Take input of rows and columns from where to split.
- Print actual matrix.
- Print the four partitions returned by function.

Your class my2D should have all the necessary function required to complete this task.

<u>Question No. 02:</u> **Marks: 40**

Write a function **void RemoveAllRepititions(int**& arr)** that takes a 2-D array "**arr**" as input and removes data repetitions in two passes. In first pass, it removes duplicate elements from each integer array. For example, integer array {1, 2, 2, 4, 4, -1} becomes {1, 2, 4, -1} after removing repetitions. In second pass, it removes repetitions if two consecutive arrays are identical. Sample run is shown below:

		
Input Array	**Array after Pass 1**	**Array after Pass 2**
	Identical consecutive elements have been removed from all rows	Identical consecutive rows removed. (First two integer arrays were identical in Pass 1)

Assume that elements in integer arrays are sorted in ascending order. Null (0) in int* array indicates end of integer arrays (delimiter in first dimension) while -1 indicates end of data in integer arrays (delimiter in second dimension). **Make sure that your program does not consume any extra memory and it should also not leak any memory.**