Programming Fundamentals

BS (CS) _Fall_2025

Lab_15 Tasks

Learning Objectives:

1. Pointer
2. Dynamic memory

Lab Tasks

Submission Instructions

1. Name each Task question as **i25XXXX_Task<NO>** **e.g.** i250000_Task1.cpp
2. Compress all **.cpp** files into a **.zip** file, and name it as *ROLLNO_SEC_LAB15* **e.g. i25XXXX_A_LAB15.**
3. Now you have to submit this zipped file on Google Classroom.
4. If you don't follow the above-mentioned submission instruction, you will be marked **zero.**
5. Plagiarism in the Lab Task will result in **zero** marks in the whole category.

Zero Tasks

Q1. Write a C++ program that takes the number of rows and columns from the user and dynamically allocates a 2D array of the specified size. Take the array values from the user, store them, and then display them. Use pointer notation only.

Q2. Write a program to create a dynamic 2D array of size 4x3. Then replace the first row with the third row, and the second row with the fourth row, by swapping their addresses only.

Lab Tasks

Note: For all questions, ensure that you properly deallocate memory and prevent dangling pointers.

Q3. Write a program in C++, in which initialize dynamic array of user given size with random values and call function get_square to square their values. Use pointer notation (pointer athematic) to access and manipulate array.

Prototype: int get_square(int* p, int size)

Q4. You are designing a feature for a small puzzle-game system where a 3×3 grid of numbers represents a rotating tile board. To simulate the rotation of the outer border tiles, write a C++ program that creates a 3×3 square matrix, initializes it with random values, and displays it. The user then enters a number n, and the program must rotate only the border elements of the matrix clockwise by n positions. Use pointer notation and pointer arithmetic to access and manipulate matrix elements. After performing the rotation, display the updated matrix.

For n=1

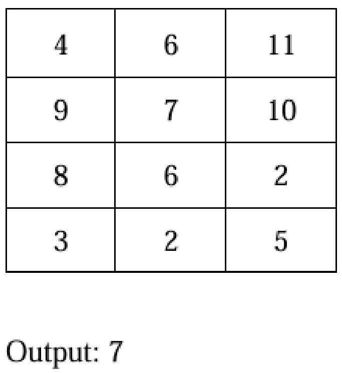
1	2	4
5	6	7
9	10	11

→

9	5	1
10	6	2
11	7	4

Q5. Write a program in C++. It takes matrix size (row, col) and create a dynamic matrix. Then take input from user for matrix and find number which is smallest in row and simultaneously largest in column. If no such number found it display a message for not found.

For example:



4	6	11
9	7	10
8	6	2
3	2	5

Output: 7

Q6. You are designing a storage manager for a program that maintains several dynamic arrays of different sizes. Initially, you create a dynamic array named root of size 3 which point to other dynamic arrays whose size is provided by the user. After filling the first three arrays, the user may choose to add more. Each time the user decides to add another array, the root array must grow by 1. Continue asking the user whether they want to create another array, and expand the root array each time the limit is reached.