Programming Fundamentals

BS (CS) _Fall_2025

Lab_12 Tasks

Learning Objectives:

1. Multi-Dimensional Array

2. Functions I (basics)

Lab Tasks

Zero Tasks

Q1.   Dry Run the code

```cpp
#include<iostream>
using namespace std;
int main()
{
    int arr[3][4]={{3,6,9,12},{2,4,6,8},{4,8,12,16}};
    for(int i=0;i<4;i++)
    {
        for(int j=0;j<3;j++)
        {
            cout<<arr[j][i]<<" ";
        }
        cout<<endl;
    }
}
```

Q2.   Write a C++ program that takes a **2×3** matrix as input from the user. Compute its transpose and store the result in a new **3×2** matrix. Display both the original and the transposed matrices.

Lab Tasks

Q3. You are working for a stock investment company, and your task is to create a program that identifies new rising stocks based on recent performance data. The user will input stock information in the form of a 5×2 matrix, you need to create a function to validate input as positive number before adding to matrix.

where:

- Each column represents one stock.
- Each row represents the stock's value for a day (over the last 5 days).
- The top value in each column represents the most recent day, and the bottom value represents the oldest day.

Your program should analyze the two stocks and determine which one is performing better according to the following analysis policy:

- If a stock's values show a strictly increasing trend over the 5 days, it is considered good.
- If both stocks show an increasing trend, compare their average values over the 5 days, the one with the higher average is better.
- If neither stock follows a fully increasing trend, determine which stock has the longest consecutive increasing pattern.
- If both have the same length of increasing sequence, select the stock that most recently follows that increasing pattern.

Display which stock is performing better based on the above criteria.

Q4. Write a program in C++. Which find the given number is Armstrong number or not.

Armstrong number is a number that equals the sum of its digits, each raised to the power of the number of digits in the number.

Example:

153 is Armstrong number, total number of digits is equal to 3.

$(1^3) + (5^3) + (3^3) = 1+125+27=153$

You need to wirte one additional funtions then main for finding total number of digits.

Q5.   Write a C++ program that includes two functions:

is_palindrome: This function accepts an integer number as a parameter and checks whether it is a palindrome or not. It returns true if the number is a palindrome, and false otherwise.

print_palindromes: This function takes two parameters as input: start (an integer) and end (an integer). It runs a loop from start to end (both inclusive) and checks whether each number is a palindrome using the is_palindrome function. If the number is a palindrome, it prints it otherwise, it skips it. This function does not return any value.

In the main function, take the start and end integer inputs and pass them to the **print_palindromes** function.