**SE Introduction To Programming**

● **Write a program to make multiplication of 2-D Matrix**

Make sure each business logic is denoted with appropriate comments and make your code interactive and represent clean and clear output on your console screen.

Adhere to the coding principles

To Accomplish this program you have to use 2-Dimensional Array concept

Firstly take the matrix input from user and display elements in Matrix format :

#include <iostream>

#include <vector>

using namespace std;

// Function to input a matrix

vector<vector<int>> inputMatrix(string name, int &rows, int &cols) {

cout << "Enter number of rows for " << name << ": ";

cin >> rows;

cout << "Enter number of columns for " << name << ": ";

cin >> cols;

vector<vector<int>> matrix(rows, vector<int>(cols));

cout << "\nEnter elements of " << name << " row-wise:\n";

for (int i = 0; i < rows; i++) {

for (int j = 0; j < cols; j++) {

cout << "Enter element [" << i + 1 << "][" << j + 1 << "]: ";

cin >> matrix[i][j];

}

}

return matrix;

}

// Function to display a matrix

void displayMatrix(const vector<vector<int>> &matrix, string name) {

cout << "\n" << name << ":\n";

for (int i = 0; i < matrix.size(); i++) {

for (int j = 0; j < matrix[i].size(); j++) {

cout << matrix[i][j] << "\t";

}

cout << endl;

}

}

// Function to multiply two matrices

vector<vector<int>> multiplyMatrices(const vector<vector<int>> &A, const vector<vector<int>> &B) {

int rowsA = A.size();

int colsA = A[0].size();

int rowsB = B.size();

int colsB = B[0].size();

vector<vector<int>> result(rowsA, vector<int>(colsB, 0));

// Multiplication logic

for (int i = 0; i < rowsA; i++) {

for (int j = 0; j < colsB; j++) {

for (int k = 0; k < colsA; k++) {

result[i][j] += A[i][k] \* B[k][j];

}

}

}

return result;

}

int main() {

cout << "MATRIX MULTIPLICATION PROGRAM" << endl;

cout << "--------------------------------" << endl;

int rowsA, colsA, rowsB, colsB;

// Input matrices

vector<vector<int>> A = inputMatrix("Matrix A", rowsA, colsA);

vector<vector<int>> B = inputMatrix("Matrix B", rowsB, colsB);

// Check multiplication condition

if (colsA != rowsB) {

cout << "\nMatrix multiplication not possible!" << endl;

cout << "Number of columns of Matrix A must equal rows of Matrix B." << endl;

return 0;

}

// Display input matrices

displayMatrix(A, "Matrix A");

displayMatrix(B, "Matrix B");

// Multiply matrices

vector<vector<int>> result = multiplyMatrices(A, B);

// Display result

displayMatrix(result, "Resultant Matrix (A x B)");

return 0;

}

**Output:**

MATRIX MULTIPLICATION PROGRAM

--------------------------------

Enter number of rows for Matrix A: 2

Enter number of columns for Matrix A: 3

Enter elements of Matrix A row-wise:

Enter element [1][1]: 1

Enter element [1][2]: 2

Enter element [1][3]: 3

Enter element [2][1]: 4

Enter element [2][2]: 5

Enter element [2][3]: 6

Enter number of rows for Matrix B: 3

Enter number of columns for Matrix B: 2

Enter elements of Matrix B row-wise:

Enter element [1][1]: 7

Enter element [1][2]: 8

Enter element [2][1]: 9

Enter element [2][2]: 10

Enter element [3][1]: 11

Enter element [3][2]: 12

Matrix A:

1 2 3

4 5 6

Matrix B:

7 8

9 10

11 12

Resultant Matrix (A x B):

58 64

139 154