

**LAB-05**

**Title: Algorithm to perform following operations on matrices: Addition, Subtraction, Multiplication, Transpose.**

**Name: Azizul Abedin Azmi**

**ID: 2022-1-60-130**

**Section: 03**

**Course Code: CSE207**

**Course Title: (Data Structures)**

**Date: 11/03/2024**

**Course Instructor:**

**Dr. Anup Kumar Paul**

**Associate Professor**

**Department of Computer Science and Engineering**

**Source Code:**

**Main.java:**

import java.util.Scanner;

public class Main {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the number of rows of matrix A: ");

        int m = scanner.nextInt();

        System.out.print("Enter the number of columns of matrix A: ");

        int n = scanner.nextInt();

        int[][] a = new int[m][n];

        System.out.println("Enter the elements of matrix A:");

        for (int i = 0; i < m; i++) {

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

                a[i][j] = scanner.nextInt();

            }

        }

        System.out.print("Enter the number of rows of matrix B: ");

        int p = scanner.nextInt();

        System.out.print("Enter the number of columns of matrix B: ");

        int q = scanner.nextInt();

        int[][] b = new int[p][q];

        System.out.println("Enter the elements of matrix B:");

        for (int i = 0; i < p; i++) {

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

                b[i][j] = scanner.nextInt();

            }

        }

        MatrixOperations matrixOps = new MatrixOperations();

        System.out.println("\nMatrix A:");

        matrixOps.displayMatrix(a);

        System.out.println("Matrix B:");

        matrixOps.displayMatrix(b);

        int[][] sum = matrixOps.add(a, b);

        System.out.println("Addition of matrices:");

        matrixOps.displayMatrix(sum);

        int[][] subtraction = matrixOps.subtract(a, b);

        System.out.println("Subtraction of matrices:");

        matrixOps.displayMatrix(subtraction);

        int[][] multiplication = matrixOps.multiply(a, b);

        System.out.println("Multiplication of matrices:");

        matrixOps.displayMatrix(multiplication);

        System.out.println("Transpose of matrix A:");

        int[][] transposeA = matrixOps.transpose(a);

        matrixOps.displayMatrix(transposeA);

        System.out.println("Transpose of matrix B:");

        int[][] transposeB = matrixOps.transpose(b);

        matrixOps.displayMatrix(transposeB);

        scanner.close();

    }

}

**MatrixOperations.java:**

package Lab05;

public class MatrixOperations {

    public int[][] add(int[][] a, int[][] b) {

        int m = a.length;

        int n = a[0].length;

        int[][] sum = new int[m][n];

        for (int i = 0; i < m; i++) {

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

                sum[i][j] = a[i][j] + b[i][j];

            }

        }

        return sum;

    }

    public int[][] subtract(int[][] a, int[][] b) {

        int m = a.length;

        int n = a[0].length;

        int[][] subtraction = new int[m][n];

        for (int i = 0; i < m; i++) {

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

                subtraction[i][j] = a[i][j] - b[i][j];

            }

        }

        return subtraction;

    }

    public int[][] multiply(int[][] a, int[][] b) {

        int m = a.length;

        int n = a[0].length;

        int p = b.length;

        int q = b[0].length;

        if (n != p) {

            throw new IllegalArgumentException("Matrices cannot be multiplied due to incompatible dimensions.");

        }

        int[][] multiplication = new int[m][q];

        for (int i = 0; i < m; i++) {

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

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

                    multiplication[i][j] += a[i][k] \* b[k][j];

                }

            }

        }

        return multiplication;

    }

    public int[][] transpose(int[][] a) {

        int m = a.length;

        int n = a[0].length;

        int[][] result = new int[n][m];

        for (int i = 0; i < m; i++) {

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

                result[j][i] = a[i][j];

            }

        }

        return result;

    }

    public void displayMatrix(int[][] matrix) {

        int m = matrix.length;

        int n = matrix[0].length;

        for (int i = 0; i < m; i++) {

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

                System.out.print(matrix[i][j] + " ");

            }

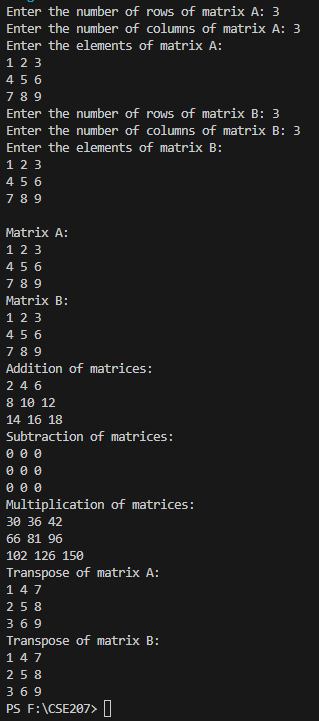
            System.out.println();

        }

    }

}

**Output:**

****