**OS LAB 1 (08/06/23)**

1.Write a c/c++ program to perform matrix operations(add,subtract,multiply,transpose)

// write a program to perform operations on matrices (addition /subtraction)

#include <stdio.h>

#include <conio.h>

void add(int matrix1[3][3], int matrix2[3][3]);

void subtract(int matrix1[3][3], int matrix2[3][3]);

void display(int matrix1[3][3], int matrix2[3][3]);

void transpose(int matrix1[3][3]);

void multiply(int matrix1[3][3], int matrix2[3][3]);

void main()

{

int choice;

int key;

int matrix1[3][3], matrix2[3][3];

printf("Enter the elements for 3X3 matrix1\n");

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

scanf("%d", &matrix1[i][j]);

}

}

printf("Enter the elements for 3X3 matrix2\n");

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

scanf("%d", &matrix2[i][j]);

}

}

while (1)

{

printf("1.Add\n2.Subtract\n3.display input matrices\n4.transpose\n5.multiply");

scanf("%d", &choice);

switch (choice)

{

case 1:

add(matrix1, matrix2);

break;

case 2:

subtract(matrix1, matrix2);

break;

case 3:

display(matrix1, matrix2);

break;

case 4:

printf("Which matrix 1 or 2?\n");

scanf("%d", &key);

if (key == 1)

{

transpose(matrix1);

}

else

transpose(matrix2);

break;

case 5:

multiply(matrix1, matrix2);

break;

default:

printf("Wrong choice!");

}

}

}

// Adding

void add(int matrix1[3][3], int matrix2[3][3])

{

int matrix3[3][3];

printf("Adding two matrices");

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

matrix3[i][j] = matrix1[i][j] + matrix2[i][j];

}

}

printf("After addition of matrix\n");

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

printf("%d\t", matrix3[i][j]);

}

}

printf("/n");

}

// subtract

void subtract(int matrix1[3][3], int matrix2[3][3])

{

int matrix3[3][3];

printf("subtracting two matrices");

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

matrix3[i][j] = matrix1[i][j] - matrix2[i][j];

}

}

printf("After addition of matrix\n");

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

printf("%d\n", matrix3[i][j]);

}

}

printf("/n");

}

// display

void display(int matrix1[3][3], int matrix2[3][3])

{

printf("Matrix 1\n");

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

printf("%d\t", matrix1[i][j]);

}

}

printf("\n");

printf("Matrix 2\n");

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

printf("%d\t", matrix2[i][j]);

}

}

printf("\n");

}

// transpose

void transpose(int matrix1[3][3])

{

int matrix4[3][3];

printf("transpose of matrix\n");

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

matrix4[j][i] = matrix1[i][j];

}

}

// result

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

printf("%d\t", matrix4[i][j]);

}

printf("\n");

}

}

// multiply

void multiply(int matrix1[3][3], int matrix2[3][3])

{

int i, j, k;

int matrix5[3][3];

for (i = 0; i < 3; i++)

{

for (j = 0; j < 3; j++)

{

matrix5[i][j] = 0;

for (k = 0; k < 3; k++)

{

matrix5[i][j] += matrix1[i][k] \* matrix2[k][j];

}

}

}

printf("after multiplication\n");

for (i = 0; i < 3; i++)

{

for (j = 0; j < 3; j++)

{

printf("%d\t", matrix5[i][j]);

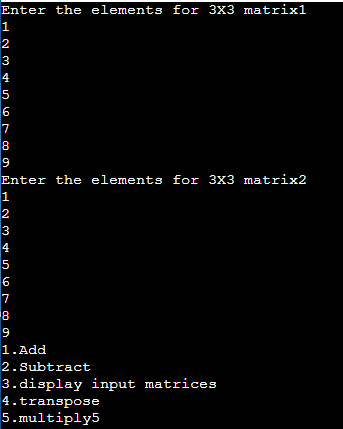
}

printf("\n");

}

}

**Output:**

****

