\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Computer Programming Lab

CEN-392

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Program 1

Code :-

#include <stdio.h>

void Addition(int arr[][10], int r1, int c1, int brr[][10], int r2, int c2)

{

    if (r1 != r2 && c1 != c2)

    {

        printf("Addition Of Two Given Matrix Not Possible!\n\n");

        return;

    }

    int ans[r1][c1];

    printf("\nAddition Answer : \n");

    for (int i = 0; i < r1; i++)

    {

        for (int j = 0; j < c1; j++)

        {

            ans[i][j] = arr[i][j] + brr[i][j];

            printf("%d ", ans[i][j]);

        }

        printf("\n");

    }

}

void Subtraction(int arr[][10], int r1, int c1, int brr[][10], int r2, int c2)

{

    if (r1 != r2 && c1 != c2)

    {

        printf("Subtraction Of Two Given Matrix Not Possible!\ns\n");

        return;

    }

    int ans[r1][c1];

    printf("\nSubtraction Answer : \n");

    for (int i = 0; i < r1; i++)

    {

        for (int j = 0; j < c1; j++)

        {

            ans[i][j] = arr[i][j] - brr[i][j];

            printf("%d ", ans[i][j]);

        }

        printf("\n");

    }

}

void Multiplication(int arr[][10], int r1, int c1, int brr[][10], int r2, int c2)

{

    if (c1 != r2)

    {

        printf("Multiplication Of Two Given Matrix Not Possible!\n\n");

        return;

    }

    int ans[r1][c2];

    printf("\nMultiplication Answer : \n");

    for (int i = 0; i < r1; i++)

    {

        for (int j = 0; j < c2; j++)

        {

            int calc = 0;

            for (int k = 0; k < c1; k++)

            {

                calc += arr[i][k] \* brr[k][j];

            }

            ans[i][j] = calc;

            printf("%d ", ans[i][j]);

        }

        printf("\n");

    }

}

int Option(int arr[][10], int r1, int c1, int brr[][10], int r2, int c2)

{

    int optn;

    printf("Enter Your Choice : ");

    scanf("%d", &optn);

    switch (optn)

    {

    case 1:

        Addition(arr, r1, c1, brr, r2, c2);

        break;

    case 2:

        Subtraction(arr, r1, c1, brr, r2, c2);

        break;

    case 3:

        Multiplication(arr, r1, c1, brr, r2, c2);

        break;

    case 4:

        return 0;

    default:

        printf("Invalid Input Try Again!\n");

    }

    return 1;

}

void Menu()

{

    printf("\n\_\_\_Matrix Operations\_\_\_\n");

    printf("1.Addition\n");

    printf("2.Subtraction\n");

    printf("3.Multiplication\n");

    printf("4.Exit\n");

}

int main()

{

    system("cls");

    while (1)

    {

        int r1, c1;

        printf("Enter The Rows And Column Of The Matrix : \n");

        scanf("%d%d", &r1, &c1);

        int arr[10][10];

        for (int i = 0; i < r1; i++)

        {

            for (int j = 0; j < c1; j++)

            {

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

            }

        }

        int r2, c2;

        printf("Enter The Rows And Column Of The Matrix : \n");

        scanf("%d%d", &r2, &c2);

        int brr[10][10];

        for (int i = 0; i < r2; i++)

        {

            for (int j = 0; j < c2; j++)

            {

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

            }

        }

    Previous:

        Menu();

        if (!Option(arr, r1, c1, brr, r2, c2))

            break;

        int c;

        printf("\nDo You Want To Work On Previos Input Matrix [y/n] : ");

        scanf(" %c", &c);

        printf("\n");

        if (c == 'Y' || c == 'y')

            goto Previous;

    }

    return 0;

}

Output :-





