```c
#include <stdio.h>
#include <stdlib.h>

void accept(int m[][10], int rows, int cols)
{
    int i, j;

    printf("Enter Matrix");
    for (i = 0; i < rows; i++)
    {
        for (j = 0; j < cols; j++)
        {
            scanf("%d", &m[i][j]);
        }
    }
}

void display(int m[10][10], int rows, int cols)
{
    int i, j;

    for (i = 0; i < rows; i++)
    {
        for (j = 0; j < cols; j++)
        {
            printf(" %d ", m[i][j]);
        }
        printf("\n");
    }
}

void addition(int m[][10], int n[0][10], int q[0][10], int rows, int cols)
{
    int i, j;
    for (i = 0; i < rows; i++)
    {
        for (j = 0; j < cols; j++)
        {
            q[i][j] = m[i][j] + n[i][j];
        }
    }
}

void multiplication(int m[][10], int n[0][10], int q[0][10], int r1, int c1,
int c2)
{

    int i, j, k;
```

```c
    for (i = 0; i < r1; i++)
    {
        for (j = 0; j < c2; j++)
        {
            for (k = 0; k < c1; k++)
                q[i][j] += m[i][k] * n[k][j];
        }
    }
}

void subtraction(int m[0][10], int n[0][10], int q[0][10], int rows, int cols)
{
    int i, j;
    for (i = 0; i < rows; i++)
    {
        for (j = 0; j < cols; j++)
        {
            q[i][j] = m[i][j] - n[i][j];
        }
    }
}
void transpose(int m[0][10], int q[0][10], int rows, int cols)
{
    int i, j;
    for (i = 0; i < rows; i++)
    {
        for (j = 0; j < cols; j++)
        {
            q[j][i] = m[i][j];
        }
    }
}

int main()
{
    int rows, cols;
    int a[10][10];
    int b[10][10];
    int c[10][10] = {};

    int r1, c1, r2, c2;
    printf("\nEnter no. of rows for first matrix: ");
    scanf("%d", &r1);
    printf("\nEnter no. of column for first matrix: ");
    scanf("%d", &c1);
    printf("\nEnter no. of rows for second matrix:");
    scanf("%d", &r2);
    printf("\nEnter no. of column for second matrix: ");
```

```c
    scanf("%d", &c2);
    accept(a, r1, c1);
    display(a, r1, c1);
    accept(b, r2, c2);
    display(b, r2, c2);
    for (;;)
    {
        int ch;
        printf("\nFollowing operations are available\n");
        printf("\n1.Addition\n 2.Subtraction\n 3.Multiplication
\n4.Transpose\n 5.Exit\n");

        printf("\nEnter which operation you want to perform:\n");
        scanf("%d", &ch);

        switch (ch)
        {
        case 1:
            if (r1 == r2 && c1 == c2)
            {
                addition(a, b, c, r1, c1);
                display(c, r2, c2);
            }
            else
                printf("\nAddition cannot be done\n");
            break;
        case 2:
            if (r1 == r2 && c1 == c2)
            {
                subtraction(a, b, c, r1, c1);
                display(c, r1, c1);
            }
            else
                printf("\nSubtraction cannot be done\n");
            break;
        case 3:
            if (c1 == r2)
            {
                multiplication(a, b, c, r1, c1, c2);
                display(c, r1, c2);
            }
            else
                printf("\nWon't work\n");
            break;

        case 4:

            transpose(a, c, r1, c1);
```

```
            display(c, c1, r1);
            break;

        case 5:
            exit(0);
            break;
        }
    }
    return 0;
}


OUTPUT:
PS D:\User\Desktop\coding> cd "d:\User\Desktop\coding\c\" ; if ($?) { gcc
matrix1.c -o matrix1 } ; if ($?) { .\matrix1 }

Enter no. of rows for first matrix: 3

Enter no. of column for first matix: 4

Enter no. of rows for second matrix: 4

Enter no. of column for second matrix: 3
Enter Matrix elements:
1
2
3
4
5
6
7
8
9
3
4
6
 1  2  3  4
 5  6  7  8
 9  3  4  6
Enter Matrix elements:
2
3
4
5
6
7
8
9
8
```

```
7
6
5
 2  3  4
 5  6  7
 8  9  8
 7  6  5

 Following operations are available:

 1.Addition
 2.Subtraction
 3.Multiplication
 4.Transpose
 5.Exit

Enter which operation you want to perform:1

Addition cannot be done

 Following operations are available:

 1.Addition
 2.Subtraction
 3.Multiplication
 4.Transpose
 5.Exit

Enter which operation you want to perform:2

Subtraction cannot be done

 Following operations are available:

 1.Addition
 2.Subtraction
 3.Multiplication
 4.Transpose
 5.Exit

Enter which operation you want to perform:3
 64  66  62
 152  162  158
 107  117  119

 Following operations are available:

 1.Addition
```

```
 2.Subtraction
 3.Multiplication
 4.Transpose
 5.Exit

Enter which operation you want to perform:4
 1  5  9
 2  6  3
 3  7  4
 4  8  6

 Following operations are available:

 1.Addition
 2.Subtraction
 3.Multiplication
 4.Transpose
 5.Exit

Enter which operation you want to perform:5
PS D:\User\Desktop\coding\c>
```