Aim

To write a C program to perform multiplication of two matrices.

Algorithm

1. Start the program.

2. Declare two 2D arrays (matrices) and another to store the result.

3. Input the number of rows and columns for both matrices.

4. Check if the matrices can be multiplied (columns of first = rows of second).

5. Read the elements of both matrices.

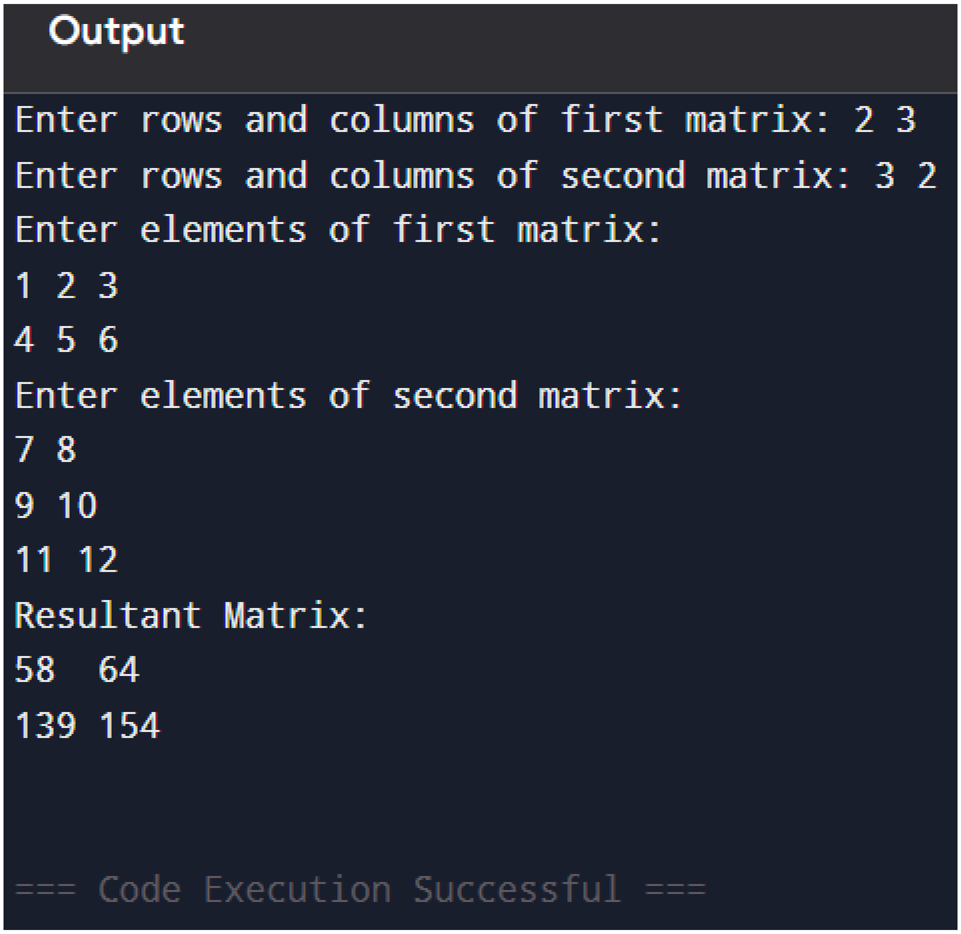
6. Perform multiplication using the formula:
   C[i][j]=∑k=0c1−1A[i][k]×B[k][j]C[i][j] = \sum_{k=0}^{c1-1} A[i][k] \times B[k][j]C[i][j]=k=0∑c1−1A[i][k]×B[k][j]
7. Display the resulting matrix.

8. End the program.

CODE:

```
#include <stdio.h>
int main() {
    int a[10][10], b[10][10], c[10][10];
    int r1, c1, r2, c2, i, j, k;

    // Input dimensions
    printf("Enter rows and columns of first matrix: ");
    scanf("%d %d", &r1, &c1);

    printf("Enter rows and columns of second matrix: ");
    scanf("%d %d", &r2, &c2);

    // Check multiplication condition
    if (c1 != r2) {
        printf("Matrix multiplication not possible!\n");
        return 0;
    }

    // Input first matrix
    printf("Enter elements of first matrix:\n");
    for (i = 0; i < r1; i++) {
        for (j = 0; j < c1; j++) {
            scanf("%d", &a[i][j]);
        }
    }

    // Input second matrix
    printf("Enter elements of second matrix:\n");
    for (i = 0; i < r2; i++) {
        for (j = 0; j < c2; j++) {
            scanf("%d", &b[i][j]);
        }
    }
    // Initialize result matrix
    for (i = 0; i < r1; i++) {
        for (j = 0; j < c2; j++) {
            c[i][j] = 0;
        }
    }

    // Perform multiplication
    for (i = 0; i < r1; i++) {
        for (j = 0; j < c2; j++) {
            for (k = 0; k < c1; k++) {
                c[i][j] += a[i][k] * b[k][j];
            }
        }
    }

    // Print result
    printf("Resultant Matrix:\n");
    for (i = 0; i < r1; i++) {
        for (j = 0; j < c2; j++) {
            printf("%d\t", c[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```

OUTPUT:



```
Enter rows and columns of first matrix: 2 3
Enter rows and columns of second matrix: 3 2
Enter elements of first matrix:
1 2 3
4 5 6
Enter elements of second matrix:
7 8
9 10
11 12
Resultant Matrix:
58  64
139 154



=== Code Execution Successful ===
```

RESULT:
The program successfully executed and displayed the multiplied matrix.