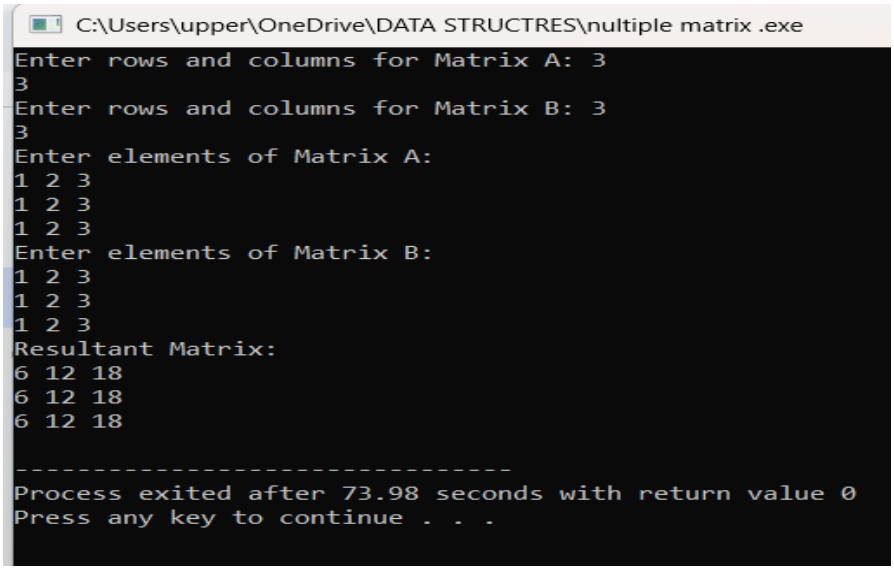
```c
#include <stdio.h>
#include <stdlib.h>
int main() {
    int r1, c1, r2, c2;
    printf("Enter rows and columns for Matrix A: ");
    scanf("%d %d", &r1, &c1);
    printf("Enter rows and columns for Matrix B: ");
    scanf("%d %d", &r2, &c2);
    if (c1 != r2) {
        printf("Error: Number of columns in Matrix A must equal number of rows in Matrix B.\n");
        return 1;
    }
    int A[r1][c1], B[r2][c2], result[r1][c2];
    printf("Enter elements of Matrix A:\n");
    for (int i = 0; i < r1; ++i)
        for (int j = 0; j < c1; ++j)
            scanf("%d", &A[i][j]);
    printf("Enter elements of Matrix B:\n");
    for (int i = 0; i < r2; ++i)
        for (int j = 0; j < c2; ++j)
            scanf("%d", &B[i][j]);
    for (int i = 0; i < r1; ++i)
        for (int j = 0; j < c2; ++j)
            result[i][j] = 0;
    for (int i = 0; i < r1; ++i)
        for (int j = 0; j < c2; ++j)
            for (int k = 0; k < c1; ++k)
                result[i][j] += A[i][k] * B[k][j];
    printf("Resultant Matrix:\n");
    for (int i = 0; i < r1; ++i) {
        for (int j = 0; j < c2; ++j)
            printf("%d ", result[i][j]);
        printf("\n");
    }
    return 0;
}
```


C:\Users\upper\OneDrive\DATA STRUCTRES\nultiple matrix .exe
Enter rows and columns for Matrix A: 3
3
Enter rows and columns for Matrix B: 3
3
Enter elements of Matrix A:
1 2 3
1 2 3
1 2 3
Enter elements of Matrix B:
1 2 3
1 2 3
1 2 3
Resultant Matrix:
6 12 18
6 12 18
6 12 18

Process exited after 73.98 seconds with return value 0
Press any key to continue . . .