

4. WAP that reads two matrices A($m \times n$) and B($p \times q$) and computes the product A and B. Read matrix A and matrix B in row major order respectively. Print both the input matrices and resultant matrix with suitable headings and output should be in matrix format only. Program must check the compatibility of orders of the matrices for multiplication. Report appropriate message in case of incompatibility.

```
#include <stdio.h>
int main () {
    int A[10][10], B[10][10], C[10][10];
    int m, n, p, q;
    int i, j, k;
    printf ("Enter rows and columns of matrix A (m n): ");
    scanf ("%d %d", &m, &n);
    printf ("Enter rows and columns of matrix B (p q): ");
    scanf ("%d %d", &p, &q);
    if (n != p) {
        printf ("\n Matrix multiplication not possible. Columns of A must equal
rows of B. \n");
        return 0;
    }
    printf ("\n Enter elements of matrix A (%d x %d): \n", m, n);
    for (i=0; i<m; i++)
        for (j=0; j<n; j++)
            scanf ("%d", &A[i][j]);
    printf ("\n Enter elements of matrix B (%d x %d): \n", p, q);
    for (i=0; i<p; i++)
        for (j=0; j<q; j++)
            scanf ("%d", &B[i][j]);
}
```

```

for (i=0 ; i<m ; i++) {
    for (j=0 ; j<q ; j++) {
        C[i][j] = 0;
        for (k=0 ; k<n ; k++) {
            C[i][j] += A[i][k] * B[k][j];
        }
    }
}

printf ("\n Matrix A:\n");
for (i=0 ; i<m ; i++) {
    for (j=0 ; j<n ; j++)
        printf ("%.4d", A[i][j]);
    printf ("\n");
}

printf ("\n Matrix B:\n");
for (i=0 ; i<p ; i++) {
    for (j=0 ; j<q ; j++)
        printf ("%.4d", B[i][j]);
    printf ("\n");
}

printf ("\n Product Matrix (AxB):\n");
for (i=0 ; i<m ; i++) {
    for (j=0 ; j<q ; j++)
        printf ("%.4d", C[i][j]);
    printf ("\n");
}

return 0;
}

```

C exp5matrices.c > ...

```
1
2 #include <stdio.h>
3
4 int main() {
5     int A[10][10], B[10][10], C[10][10];
6     int m, n, p, q;
7     int i, j, k;
8
9     // Input dimensions of matrix A
10    printf("Enter rows and columns of Matrix A (m n): ");
11    scanf("%d %d", &m, &n);
12
13    // Input dimensions of matrix B
14    printf("Enter rows and columns of Matrix B (p q): ");
15    scanf("%d %d", &p, &q);
16
17    // Check compatibility
18    if(n != p) {
19        printf("\nMatrix multiplication not possible. Columns of A must equal rows of B.\n");
20        return 0;
21    }
22
23    // Input elements of matrix A
24    printf("\nEnter elements of Matrix A (%d x %d):\n", m, n);
25    for(i = 0; i < m; i++)
26        for(j = 0; j < n; j++)
27            scanf("%d", &A[i][j]);
28
29    // Input elements of matrix B
30    printf("\nEnter elements of Matrix B (%d x %d):\n", p, q);
31    for(i = 0; i < p; i++)
32        for(j = 0; j < q; j++)
33            scanf("%d", &B[i][j]);
```

```
35 // Multiply matrices
36 for(i = 0; i < m; i++) {
37     for(j = 0; j < q; j++) {
38         C[i][j] = 0;
39         for(k = 0; k < n; k++) {
40             C[i][j] += A[i][k] * B[k][j];
41         }
42     }
43 }
44
45 // Print Matrix A
46 printf("\nMatrix A:\n");
47 for(i = 0; i < m; i++) {
48     for(j = 0; j < n; j++)
49         printf("%4d", A[i][j]);
50     printf("\n");
51 }
52
53 // Print Matrix B
54 printf("\nMatrix B:\n");
55 for(i = 0; i < p; i++) {
56     for(j = 0; j < q; j++)
57         printf("%4d", B[i][j]);
58     printf("\n");
59 }
60
61 // Print Resultant Matrix C
62 printf("\nProduct Matrix (A x B):\n");
63 for(i = 0; i < m; i++) {
64     for(j = 0; j < q; j++)
65         printf("%4d", C[i][j]);
66     printf("\n");
```

```
67     }
68
69     return 0;
70 }
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE PORTS

Enter rows and columns of Matrix A (m n): 3 2

Enter rows and columns of Matrix B (p q): 2 3

Enter elements of Matrix A (3 x 2):

2 3

4 5

6 7

Enter elements of Matrix B (2 x 3):

3 5 6

2 5 0

Matrix A:

2 3

4 5

6 7

Matrix B:

3 5 6

2 5 0

Product Matrix (A x B):

12 25 12

22 45 24

32 65 36

PS C:\Users\abiga\OneDrive\Desktop\Absproj> █