



PRACTICA 10: **PRODUCTO DE MATRICES**

ESTUDIANTE: **EFRAIN ROBLES PULIDO**

CODIGO: **221350095**

NOMBRE DE LA MATERIA: **SEMINARIO DE SOLUCION DE PROBLEMAS DE
PROGRAMACION**

SECCIÓN: **D67**

Descripción:

Objetivo de la practica

Hacer un programa para calcular el producto de 2 matrices (A·B) de orden MxN y NxP respectivamente, guardar el resultado en la matriz C de orden MxP. Los valores de las matrices A y B, pueden ser con números aleatorios o si lo prefieres se pueden solicitar que los capture el usuario. Debe mostrar además de las 3 matrices, el desglose de las operaciones realizadas.

Fundamentación teórica

La multiplicación de matrices consiste en combinar linealmente dos o más matrices mediante la adición de sus elementos dependiendo de su situación dentro de la matriz origen respetando el orden de los factores. La multiplicación de dos matrices es unificar las matrices en una sola matriz mediante la multiplicación y suma de los elementos de las filas y columnas de las matrices origen teniendo en cuenta el orden de los factores.

El proceso de multiplicación de una matriz es la siguiente:

1. Verifica que el número de filas de la primera matriz es igual al número de columnas de la segunda matriz.
2. De la primera matriz: toma una **fila i**
3. De la segunda matriz: toma una **columna j**
4. **Multiplica** cada un elemento de cada fila y de cada columna, en orden, para después sumar los productos.
5. Coloca el resultado en la posición **(i, j)** en la matriz resultante.
6. Regresa al paso 2, hasta terminar con toda la matriz.

Análisis del problema

Se deberá de introducir los valores de los tamaños de las dos matrices, en que se introducirá el tamaño de las fila (M) de la matriz A, el tamaño de la columna de la matriz A como el de la fila de la matriz B (N) y el tamaño de las columnas de la matriz B (P), después se usará 2 ciclos for para introducir los valores en cada matriz de manera aleatoria, así como para mostrar la matriz resultante del producto de matrices, y para realizar las operaciones se utilizara 3 ciclos for en donde se ira cambiando las posiciones de las matrices en cada operación para tener el producto de sumas, así como ciclos for para mostrar el desglose de las operaciones.

Datos de entrada y precondiciones

- Tamaño de la matriz de A y B – Filas y Columnas

Datos o elementos de salida

- Matriz C – matriz resultante

Desarrollo:

Procedimiento en lenguaje natural

Primero deberemos de introducir los tamaños deseados de las matrices para después introducir valores en las matrices de manera aleatoria, con la función rand de 0 a 10, después procedemos a mostrar las matrices A y B con el uso de ciclos for anidados para acceder a ellos, al igual con la matriz resultante de C, des pues de hacer la operación. A continuación, procederemos a utilizar 3 ciclos for anidados para hacer la operación de productos de matrices, el primero correrá el de la fila M, el segundo for será de la columna P y el tercero será de la N, en donde en ese ciclo for anidado estará la operación de la acumulación se la suma de los productos de las matrices de acuerdo con su posición en la matriz, en donde se hará la multiplicación de la fila de la matriz A con la columna de la matriz B, para sumar los productos que serán los valores en la matriz C. Además de mostrar el desglose de las operaciones utilizadas para la matriz C.

Algoritmo

Pseudocódigo:

Inicio

desde (m \leftarrow 0; m<M; inc m) //Producto de matrices (MxN)x(NxP) = (MxP)

desde (p \leftarrow 0; p<P; inc p)

inicio

sum \leftarrow 0

desde (n \leftarrow 0; n<N; inc n)

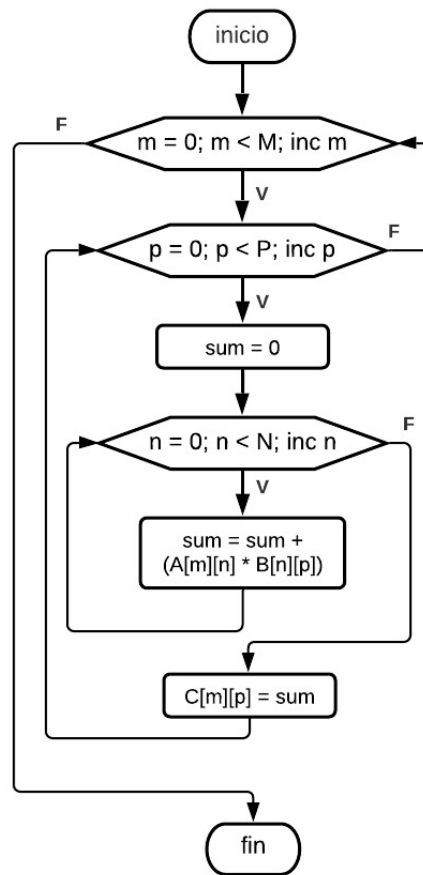
sum \leftarrow sum + (A1[m][n] * B1[n][p])

P[m][p] \leftarrow sum

fin

Fin

Diagrama de flujo



Código fuente del programa en lenguaje C

//Efrain Robles Pulido

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

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

```
#include <time.h>
```

```
void main(){//Practica 10: Producto de Matrices
```

```
    int M, N, P;
```

```
    printf("Dame valor de la fila para la matriz A:\n");
```

```
    scanf("%d",&M);
```

```
    printf("\nDame valor de la columna para la matriz A,\n como para la fila de la matriz B:\n");
```

```
    scanf("%d",&N);
```

```
printf("\nDame valor de la columna de la matriz B:\n");
```

```
scanf("%d",&P);
```

```
int sum, m, n, p, A[M][N], B[N][P],C[M][P];
```

```
srand(time(NULL)); //inicializador de numeros aleatorios
```

```
for(m=0;m<M;m++)//Llenado de MxN
```

```
    for(n=0;n<N;n++)
```

```
        A[m][n]=rand()%10;
```

```
for(n=0;n<N;n++)//Llenado de NxP
```

```
    for(p=0;p<P;p++)
```

```
        B[n][p]=rand()%10;
```

```
printf("\n  Matriz MxN\n");
```

```
for(m=0;m<M;m++){//Impresion de MxN
```

```
    for(n=0;n<N;n++)
```

```
        printf("%3d ",A[m][n]);
```

```
    printf("\n");
```

```
}
```

```
printf("\n  Matriz NxP\n");
```

```
for(n=0;n<N;n++){//Impresion de NxP
```

```
    for(p=0;p<P;p++)
```

```
        printf("%3d ",B[n][p]);
```

```
    printf("\n");
```

```
}
```

```
printf("\n");
```

```
for(m=0;m<M;m++){//Producto de matrices (MxN)x(NxP) = (MxP)
```

```
    for(p=0;p<P;p++){
```

```

printf("C%d%d = ",m,p);

sum=0;

for(n=0;n<N;n++){

    sum=sum+(A[m][n]*B[n][p]);

    printf("A%d*d*B%d%d %c ",m,n,n,p,(n<N-1)?'+':'=');//Desgloce de operaciones
}

for(n=0;n<N;n++)//Desgloce de operaciones

    printf("%d * %d %c ",A[m][n],B[n][p],(n<N-1)?'+':'=');

for(n=0;n<N;n++)//Desgloce de operaciones

    printf("%3d %c",A[m][n]*B[n][p],(n<N-1)?'+':'=');


C[m][p]=sum;

printf(" %d\n",C[m][p]); //Desgloce de operaciones
}

}

printf("\n  Matriz MxP\n");

for(m=0;m<M;m++){//Impresion de MxP

    for(p=0;p<P;p++)

        printf(" %3d ",C[m][p]);

    printf("\n");

}

}

```

Resultados obtenidos:

```
//Efrain Robles Pulido
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

void main() { //Practica 10: Producto de Matrices
    int M, N, P;

    printf("Dame valor de la fila para la matriz A:\n");
    scanf("%d", &M);
    printf("\nDame valor de la columna para la matriz A,\n como para la fila de la matriz B:\n");
    scanf("%d", &N);
    printf("\nDame valor de la columna de la matriz B:\n");
    scanf("%d", &P);

    int sum, m, n, p, A[M][N], B[N][P], C[M][P];
    srand(time(NULL)); //inicializador de numeros aleatorios

    for(m=0; m<M; m++) //llenado de MxN
        for(n=0; n<N; n++)
            A[m][n]=rand()%10;

    for(n=0; n<N; n++) //llenado de NxP
        for(p=0; p<P; p++)
            B[n][p]=rand()%10;

    printf("\n Matriz MxN\n");
    for(m=0; m<M; m++) { //Impresion de MxN
        for(n=0; n<N; n++)
            printf("%3d ", A[m][n]);
        printf("\n");
    }

    printf("\n Matriz NxP\n");
    for(n=0; n<N; n++) { //Impresion de NxP
        for(p=0; p<P; p++)
            printf("%3d ", B[n][p]);
        printf("\n");
    }

    printf("\n");
    for(m=0; m<M; m++) { //Producto de matrices (MxN)x(NxP) = (MxP)
        for(p=0; p<P; p++) {
            printf("C%d%d = ", m, p);
            sum=0;
            for(n=0; n<N; n++) {
                sum=sum+(A[m][n]*B[n][p]);
                printf("A%d%d*B%d%d %c ", m, n, n, p, (n<N-1)?'+':'='); //Desglose de operaciones
            }
            for(n=0; n<N; n++) //Desglose de operaciones
                printf("%d * %d %c ", A[m][n], B[n][p], (n<N-1)?'+':'=');

            for(n=0; n<N; n++) //Desglose de operaciones
                printf("%3d %c", A[m][n]*B[n][p], (n<N-1)?'+':'=');

            C[m][p]=sum;
            printf(" %d\n", C[m][p]); //Desglose de operaciones
        }
    }

    printf("\n Matriz MxP\n");
    for(m=0; m<M; m++) { //Impresion de MxP
        for(p=0; p<P; p++)
            printf(" %3d ", C[m][p]);
        printf("\n");
    }
}
```

```

Dame valor de la fila para la matriz A:
3

Dame valor de la columna para la matriz A,
como para la fila de la matriz B:
3

Dame valor de la columna de la matriz B:
3

Matriz MxN
6 3 1
0 9 4
3 9 1

Matriz NxP
2 4 7
2 8 9
1 3 1

C00 = A00*B00 + A01*B10 + A02*B20 = 6 * 2 + 3 * 2 + 1 * 1 = 12 + 6 + 1 = 19
C01 = A00*B01 + A01*B11 + A02*B21 = 6 * 4 + 3 * 8 + 1 * 3 = 24 + 24 + 3 = 51
C02 = A00*B02 + A01*B12 + A02*B22 = 6 * 7 + 3 * 9 + 1 * 1 = 42 + 27 + 1 = 70
C10 = A10*B00 + A11*B10 + A12*B20 = 0 * 2 + 9 * 2 + 4 * 1 = 0 + 18 + 4 = 22
C11 = A10*B01 + A11*B11 + A12*B21 = 0 * 4 + 9 * 8 + 4 * 3 = 0 + 72 + 12 = 84
C12 = A10*B02 + A11*B12 + A12*B22 = 0 * 7 + 9 * 9 + 4 * 1 = 0 + 81 + 4 = 85
C20 = A20*B00 + A21*B10 + A22*B20 = 3 * 2 + 9 * 2 + 1 * 1 = 6 + 18 + 1 = 25
C21 = A20*B01 + A21*B11 + A22*B21 = 3 * 4 + 9 * 8 + 1 * 3 = 12 + 72 + 3 = 87
C22 = A20*B02 + A21*B12 + A22*B22 = 3 * 7 + 9 * 9 + 1 * 1 = 21 + 81 + 1 = 103

Matriz MxP
19 51 70
22 84 85
25 87 103

Process returned 3 (0x3) execution time : 1.870 s
Press any key to continue.

```

```

Dame valor de la fila para la matriz A:
4

Dame valor de la columna para la matriz A,
como para la fila de la matriz B:
4

Dame valor de la columna de la matriz B:
4

Matriz MxN
2 8 1 9
1 8 2 1
3 2 4 3
3 9 4 3

Matriz NxP
7 9 0 5
4 8 8 1
2 8 9 5
2 4 2 6

C00 = A00*B00 + A01*B10 + A02*B20 + A03*B30 = 2 * 7 + 8 * 4 + 1 * 2 + 9 * 2 = 14 + 32 + 2 + 18 = 66
C01 = A00*B01 + A01*B11 + A02*B21 + A03*B31 = 2 * 9 + 8 * 8 + 1 * 8 + 9 * 4 = 18 + 64 + 8 + 36 = 126
C02 = A00*B02 + A01*B12 + A02*B22 + A03*B32 = 2 * 0 + 8 * 8 + 1 * 9 + 9 * 2 = 0 + 64 + 9 + 18 = 91
C03 = A00*B03 + A01*B13 + A02*B23 + A03*B33 = 2 * 5 + 8 * 1 + 1 * 5 + 9 * 6 = 10 + 8 + 5 + 54 = 77
C10 = A10*B00 + A11*B10 + A12*B20 + A13*B30 = 1 * 7 + 8 * 4 + 2 * 2 + 1 * 2 = 7 + 32 + 4 + 2 = 45
C11 = A10*B01 + A11*B11 + A12*B21 + A13*B31 = 1 * 9 + 8 * 8 + 2 * 8 + 1 * 4 = 9 + 64 + 16 + 4 = 93
C12 = A10*B02 + A11*B12 + A12*B22 + A13*B32 = 1 * 0 + 8 * 8 + 2 * 9 + 1 * 2 = 0 + 64 + 18 + 2 = 84
C13 = A10*B03 + A11*B13 + A12*B23 + A13*B33 = 1 * 5 + 8 * 1 + 2 * 5 + 1 * 6 = 5 + 8 + 10 + 6 = 29
C20 = A20*B00 + A21*B10 + A22*B20 + A23*B30 = 3 * 7 + 2 * 4 + 4 * 2 + 3 * 2 = 21 + 8 + 8 + 6 = 43
C21 = A20*B01 + A21*B11 + A22*B21 + A23*B31 = 3 * 9 + 2 * 8 + 4 * 8 + 3 * 4 = 27 + 16 + 32 + 12 = 87
C22 = A20*B02 + A21*B12 + A22*B22 + A23*B32 = 3 * 0 + 2 * 8 + 4 * 9 + 3 * 2 = 0 + 16 + 36 + 6 = 58
C23 = A20*B03 + A21*B13 + A22*B23 + A23*B33 = 3 * 5 + 2 * 1 + 4 * 5 + 3 * 6 = 15 + 2 + 20 + 18 = 55
C30 = A30*B00 + A31*B10 + A32*B20 + A33*B30 = 3 * 7 + 9 * 4 + 4 * 2 + 3 * 2 = 21 + 36 + 8 + 6 = 71
C31 = A30*B01 + A31*B11 + A32*B21 + A33*B31 = 3 * 9 + 9 * 8 + 4 * 8 + 3 * 4 = 27 + 72 + 32 + 12 = 143
C32 = A30*B02 + A31*B12 + A32*B22 + A33*B32 = 3 * 0 + 9 * 8 + 4 * 9 + 3 * 2 = 0 + 72 + 36 + 6 = 114
C33 = A30*B03 + A31*B13 + A32*B23 + A33*B33 = 3 * 5 + 9 * 1 + 4 * 5 + 3 * 6 = 15 + 9 + 20 + 18 = 62

Matriz MxP
66 126 91 77
45 93 84 29
43 87 58 55
71 143 114 62

Process returned 4 (0x4) execution time : 4.541 s
Press any key to continue.

```



```

Dame valor de la fila para la matriz A:
4

Dame valor de la columna para la matriz A,
como para la fila de la matriz B:
3

Dame valor de la columna de la matriz B:
2

Matriz MxN
3  1  4
2  5  6
5  3  9
8  6  3

Matriz NxP
2  2
7  5
0  9

C00 = A00*B00 + A01*B10 + A02*B20 = 3 * 2 + 1 * 7 + 4 * 0 = 6 + 7 + 0 = 13
C01 = A00*B01 + A01*B11 + A02*B21 = 3 * 2 + 1 * 5 + 4 * 9 = 6 + 5 + 36 = 47
C10 = A10*B00 + A11*B10 + A12*B20 = 2 * 2 + 5 * 7 + 6 * 0 = 4 + 35 + 0 = 39
C11 = A10*B01 + A11*B11 + A12*B21 = 2 * 2 + 5 * 5 + 6 * 9 = 4 + 25 + 54 = 83
C20 = A20*B00 + A21*B10 + A22*B20 = 5 * 2 + 3 * 7 + 9 * 0 = 10 + 21 + 0 = 31
C21 = A20*B01 + A21*B11 + A22*B21 = 5 * 2 + 3 * 5 + 9 * 9 = 10 + 15 + 81 = 106
C30 = A30*B00 + A31*B10 + A32*B20 = 8 * 2 + 6 * 7 + 3 * 0 = 16 + 42 + 0 = 58
C31 = A30*B01 + A31*B11 + A32*B21 = 8 * 2 + 6 * 5 + 3 * 9 = 16 + 30 + 27 = 73

Matriz MxP
13  47
39  83
31  106
58  73

Process returned 4 (0x4)   execution time : 12.078 s
Press any key to continue.

```

Conclusión

En esta practica fue interesante utilizar los ciclos for para poder hacer el producto de matrices, ya que tuve que hacer el arreglo de ciclos anidados para cumplir con las propiedades del producto de matrices. Como se hizo anteriormente, solo tuve que adaptarlo en donde puedo introducir el tamaño de las matrices como el usuario desee, además de agregarle los pasos de las funciones de impresión de las operaciones hechas, en donde se tuvo que agregar ciclos for para que se repitiera las veces necesarias las operaciones.