

Ejercicios de matriz en C

1. Suma dos matrices 2x2.

```
#include <stdio.h>

int main() {
    int A[2][2] = {{1, 2}, {3, 4}};
    int B[2][2] = {{5, 6}, {7, 8}};
    int C[2][2];

    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2; j++)
            C[i][j] = A[i][j] + B[i][j];

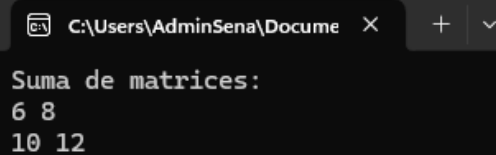
    printf("Suma de matrices:\n");
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++)
            printf("%d ", C[i][j]);
        printf("\n");
    }
    return 0;
}
```

```
#include <stdio.h>

int main() {
    int A[2][2] = {{1, 2}, {3, 4}};
    int B[2][2] = {{5, 6}, {7, 8}};
    int C[2][2];

    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2; j++)
            C[i][j] = A[i][j] + B[i][j];

    printf("Suma de matrices:\n");
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++)
            printf("%d ", C[i][j]);
        printf("\n");
    }
    return 0;
}
```



```
C:\Users\AdminSena\Docume
Suma de matrices:
6 8
10 12
```

2. Intercambiar las filas por las columnas de una matriz 2x2

```

#include <stdio.h>

int main() {
    int A[2][2] = {{1, 2}, {3, 4}}; // Matriz original
    int T[2][2]; // Matriz transpuesta

    // Transposición
    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2; j++)
            T[j][i] = A[i][j]; // Intercambiamos filas por columnas

    // Imprimir la matriz transpuesta
    printf("Matriz transpuesta:\n");
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++)
            printf("%d ", T[i][j]);
        printf("\n");
    }
    return 0;
}

```

```

#include <stdio.h>

int main() {
    int A[2][2] = {{1, 2}, {3, 4}}; // Matriz original
    int T[2][2]; // Matriz transpuesta

    // Transposición
    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2; j++)
            T[j][i] = A[i][j]; // Intercambiamos filas por columnas

    // Imprimir la matriz transpuesta
    printf("Matriz transpuesta:\n");
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++)
            printf("%d ", T[i][j]);
        printf("\n");
    }
    return 0;
}

```

C:\Users\AdminSena\Docume X + v

Matriz transpuesta:

1 3
2 4

3. Multiplica dos matrices 2x2.

```
#include <stdio.h>

int main() {
    int A[2][2] = {{1, 2}, {3, 4}};
    int B[2][2] = {{5, 6}, {7, 8}};
    int C[2][2] = {{0, 0}, {0, 0}};

    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2; j++)
            for (int k = 0; k < 2; k++)
                C[i][j] += A[i][k] * B[k][j];

    printf("Multiplicación de matrices:\n");
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++)
            printf("%d ", C[i][j]);
        printf("\n");
    }
    return 0;
}
```

```
#include <stdio.h>

int main() {
    int A[2][2] = {{1, 2}, {3, 4}};
    int B[2][2] = {{5, 6}, {7, 8}};
    int C[2][2] = {{0, 0}, {0, 0}};

    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2; j++)
            for (int k = 0; k < 2; k++)
                C[i][j] += A[i][k] * B[k][j];

    printf("Multiplicación de matrices:\n");
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++)
            printf("%d ", C[i][j]);
        printf("\n");
    }
    return 0;
}
```

C:\Users\AdminSena\Docume X + v

```
Multiplicaci3n de matrices:
19 22
43 50
```

4. Calcular el determinante de una matriz 2x2. fórmula:

$$\det(A) = ad - bc$$

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

```
int main() {  
    int A[2][2] = {{1, 2}, {3, 4}}; // Matriz 2x2  
    int determinant;  
  
    // Cálculo del determinante  
    determinant = A[0][0] * A[1][1] - A[0][1] * A[1][0];  
  
    // Imprimir el resultado  
    printf("Determinante: %d\n", determinant);  
    return 0;  
}
```

```
#include <stdio.h>  
  
int main() {  
    int A[2][2] = {{1, 2}, {3, 4}}; // Matriz 2x2  
    int determinant;  
  
    // Cálculo del determinante  
    determinant = A[0][0] * A[1][1] - A[0][1] * A[1][0];  
  
    // Imprimir el resultado  
    printf("Determinante: %d\n", determinant);  
    return 0;  
}
```

C:\Users\AdminSena\Docume X + v

Determinante: -2

5. Encuentra el valor máximo en una matriz 2x2.

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

```
int main() {  
    int A[2][2] = {{1, 5}, {3, 2}}; // Matriz 2x2  
    int max = A[0][0]; // Inicializar max con el primer elemento  
  
    // Recorrer la matriz para encontrar el máximo  
    for (int i = 0; i < 2; i++) {
```

```
        for (int j = 0; j < 2; j++) {
            if (A[i][j] > max) {
                max = A[i][j]; // Actualizar max si encontramos un valor mayor
            }
        }
    }
}

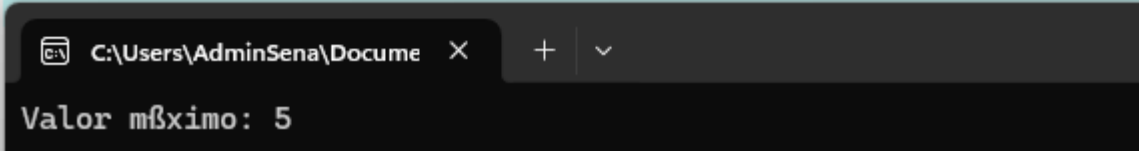
// Imprimir el valor máximo
printf("Valor máximo: %d\n", max);
return 0;
}

#include <stdio.h>

int main() {
    int A[2][2] = {{1, 5}, {3, 2}}; // Matriz 2x2
    int max = A[0][0]; // Inicializar max con el primer elemento

    // Recorrer la matriz para encontrar el máximo
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++) {
            if (A[i][j] > max) {
                max = A[i][j]; // Actualizar max si encontramos un valor mayor
            }
        }
    }

    // Imprimir el valor máximo
    printf("Valor máximo: %d\n", max);
    return 0;
}
```



The screenshot shows a Windows command prompt window with a single tab titled 'C:\Users\AdminSena\Docume'. The command prompt displays the output 'Valor máximo: 5'.

6. Cuenta cuántos números positivos hay en una matriz 2x2.

```
#include <stdio.h>

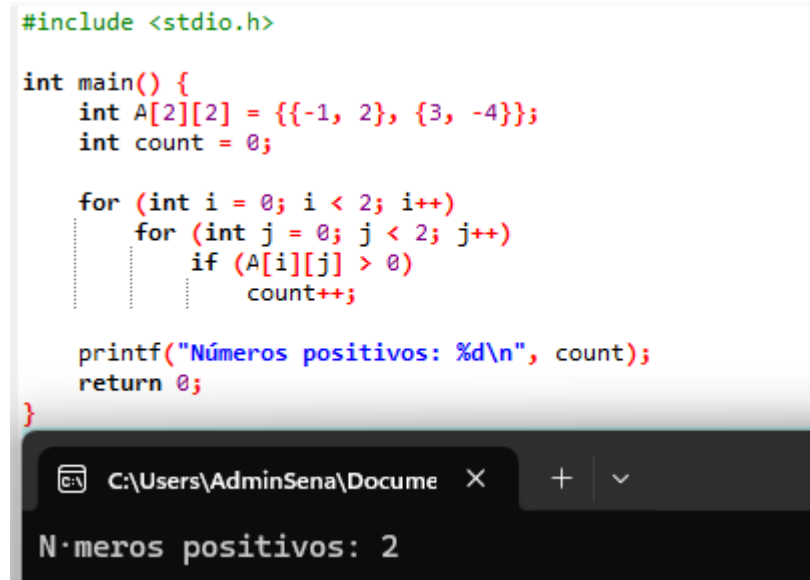
int main() {
    int A[2][2] = {{-1, 2}, {3, -4}};
    int count = 0;

    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2; j++)
            if (A[i][j] > 0)
                count++;
}
```

```

printf("Números positivos: %d\n", count);
return 0;
}

```



The image shows a C program in a code editor and its output in a terminal window. The code defines a 2x2 matrix A with values {{-1, 2}, {3, -4}} and counts the number of positive elements. The terminal output shows "N.meros positivos: 2".

```

#include <stdio.h>

int main() {
    int A[2][2] = {{-1, 2}, {3, -4}};
    int count = 0;

    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2; j++)
            if (A[i][j] > 0)
                count++;

    printf("Números positivos: %d\n", count);
    return 0;
}

```

Output: N.meros positivos: 2

7. Calcula la inversa de una matriz 2x2.

```

#include <stdio.h>

int main() {
    int A[2][2] = {{1, 2}, {3, 4}};
    int determinant = A[0][0] * A[1][1] - A[0][1] * A[1][0];
    float inv[2][2];

    if (determinant != 0) {
        inv[0][0] = (float)A[1][1] / determinant;
        inv[0][1] = -(float)A[0][1] / determinant;
        inv[1][0] = -(float)A[1][0] / determinant;
        inv[1][1] = (float)A[0][0] / determinant;

        printf("Matriz inversa:\n");
        for (int i = 0; i < 2; i++) {
            for (int j = 0; j < 2; j++)
                printf("%.2f ", inv[i][j]);
            printf("\n");
        }
    } else {
        printf("La matriz no es invertible.\n");
    }
}

```

```

    }
    return 0;
}

```

```

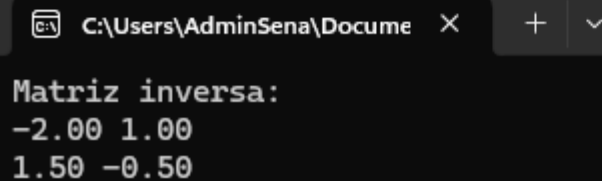
#include <stdio.h>

int main() {
    int A[2][2] = {{1, 2}, {3, 4}};
    int determinant = A[0][0] * A[1][1] - A[0][1] * A[1][0];
    float inv[2][2];

    if (determinant != 0) {
        inv[0][0] = (float)A[1][1] / determinant;
        inv[0][1] = -(float)A[0][1] / determinant;
        inv[1][0] = -(float)A[1][0] / determinant;
        inv[1][1] = (float)A[0][0] / determinant;

        printf("Matriz inversa:\n");
        for (int i = 0; i < 2; i++) {
            for (int j = 0; j < 2; j++)
                printf("%.2f ", inv[i][j]);
            printf("\n");
        }
    } else {
        printf("La matriz no es invertible.\n");
    }
    return 0;
}

```



```

C:\Users\AdminSena\Docume
Matriz inversa:
-2.00 1.00
1.50 -0.50

```

8. Genera una matriz identidad 2x2.: Una matriz identidad es una matriz cuadrada en la que los elementos de la diagonal principal son 1, y todos los demás elementos son 0. Para una matriz de tamaño 2x2, la matriz identidad se representa de la siguiente manera:

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

```

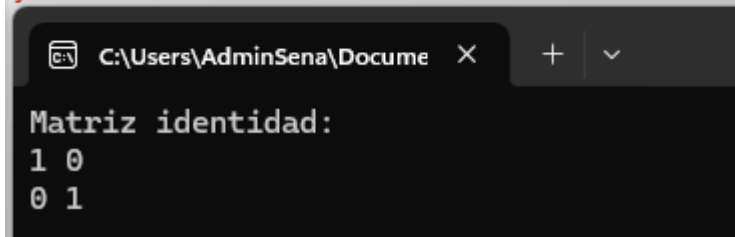
int main() {
    int I[2][2] = {{1, 0}, {0, 1}};

    printf("Matriz identidad:\n");
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++)
            printf("%d ", I[i][j]);
        printf("\n");
    }
}

```

```
}  
return 0;  
}
```

```
#include <stdio.h>  
  
int main() {  
    int I[2][2] = {{1, 0}, {0, 1}};  
  
    printf("Matriz identidad:\n");  
    for (int i = 0; i < 2; i++) {  
        for (int j = 0; j < 2; j++)  
            printf("%d ", I[i][j]);  
        printf("\n");  
    }  
    return 0;  
}
```



The screenshot shows a Windows command prompt window with the title bar "C:\Users\AdminSena\Docume". The output of the program is displayed in a monospaced font: "Matriz identidad:" followed by two lines of "1 0" and "0 1".

9. Imprime una matriz 2x2.

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

```
int main() {  
    int A[2][2] = {{1, 2}, {3, 4}};  
  
    printf("Matriz:\n");  
    for (int i = 0; i < 2; i++) {  
        for (int j = 0; j < 2; j++)  
            printf("%d ", A[i][j]);  
        printf("\n");  
    }  
    return 0;  
}
```



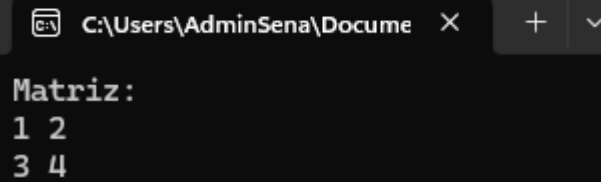
```

#include <stdio.h>

int main() {
    int A[2][2] = {{1, 2}, {3, 4}};

    printf("Matriz:\n");
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++)
            printf("%d ", A[i][j]);
        printf("\n");
    }
    return 0;
}

```



```

C:\Users\AdminSena\Docume
Matriz:
1 2
3 4

```

10. Multiplica todos los elementos de una matriz 2x2 por un escalar.

```

#include <stdio.h>

int main() {
    int A[2][2] = {{1, 2}, {3, 4}};
    int escalar = 2;
    int B[2][2];

    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2; j++)
            B[i][j] = A[i][j] * escalar;

    printf("Matriz multiplicada por escalar:\n");
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++)
            printf("%d ", B[i][j]);
        printf("\n");
    }
    return 0;
}

```

```
#include <stdio.h>

int main() {
    int A[2][2] = {{1, 2}, {3, 4}};
    int escalar = 2;
    int B[2][2];

    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2; j++)
            B[i][j] = A[i][j] * escalar;

    printf("Matriz multiplicada por escalar:\n");
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++)
            printf("%d ", B[i][j]);
        printf("\n");
    }
    return 0;
}
```

C:\Users\AdminSena\Docume X + v

Matriz multiplicada por escalar:
2 4
6 8