

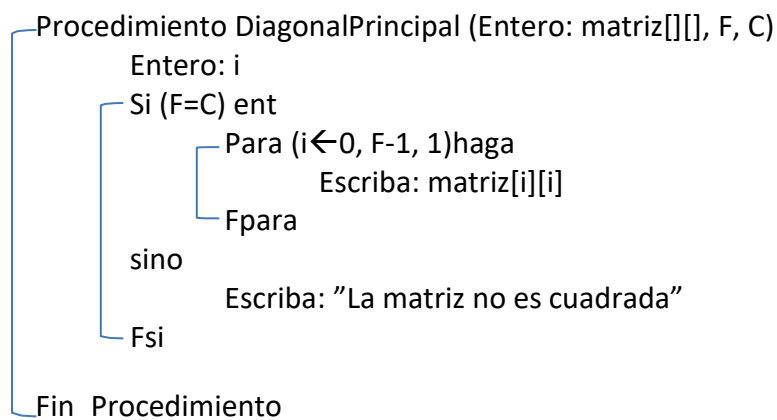


TIPO DE MATRICES

Realice algoritmos para los siguientes tipos de matrices (tenga en cuenta, cuando se deba verificar si una matriz es cuadrada, es decir que tenga el mismo número de filas y columnas):

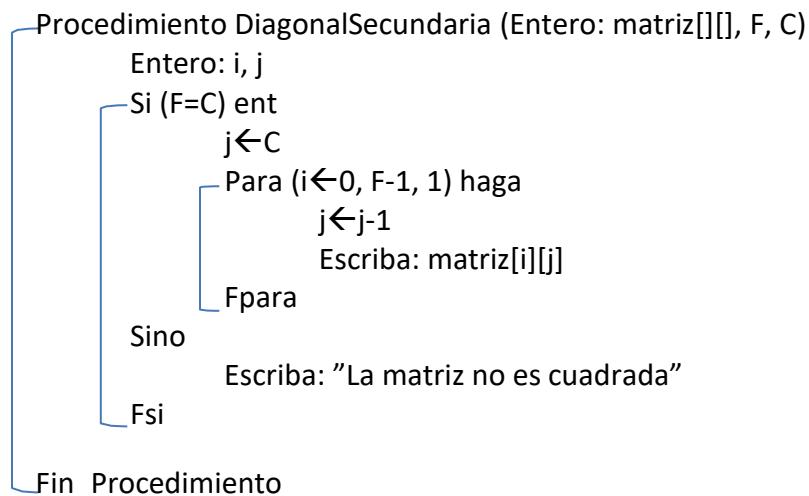
1. **DIAGONAL PRINCIPAL:** Realice un subprograma que muestre los elementos de la Diagonal Principal de una Matriz.

1	2	3
4	5	6
7	8	9



2. **DIAGONAL SECUNDARIA:** Realice un subprograma que muestre los elementos de la Diagonal Secundaria de una Matriz.

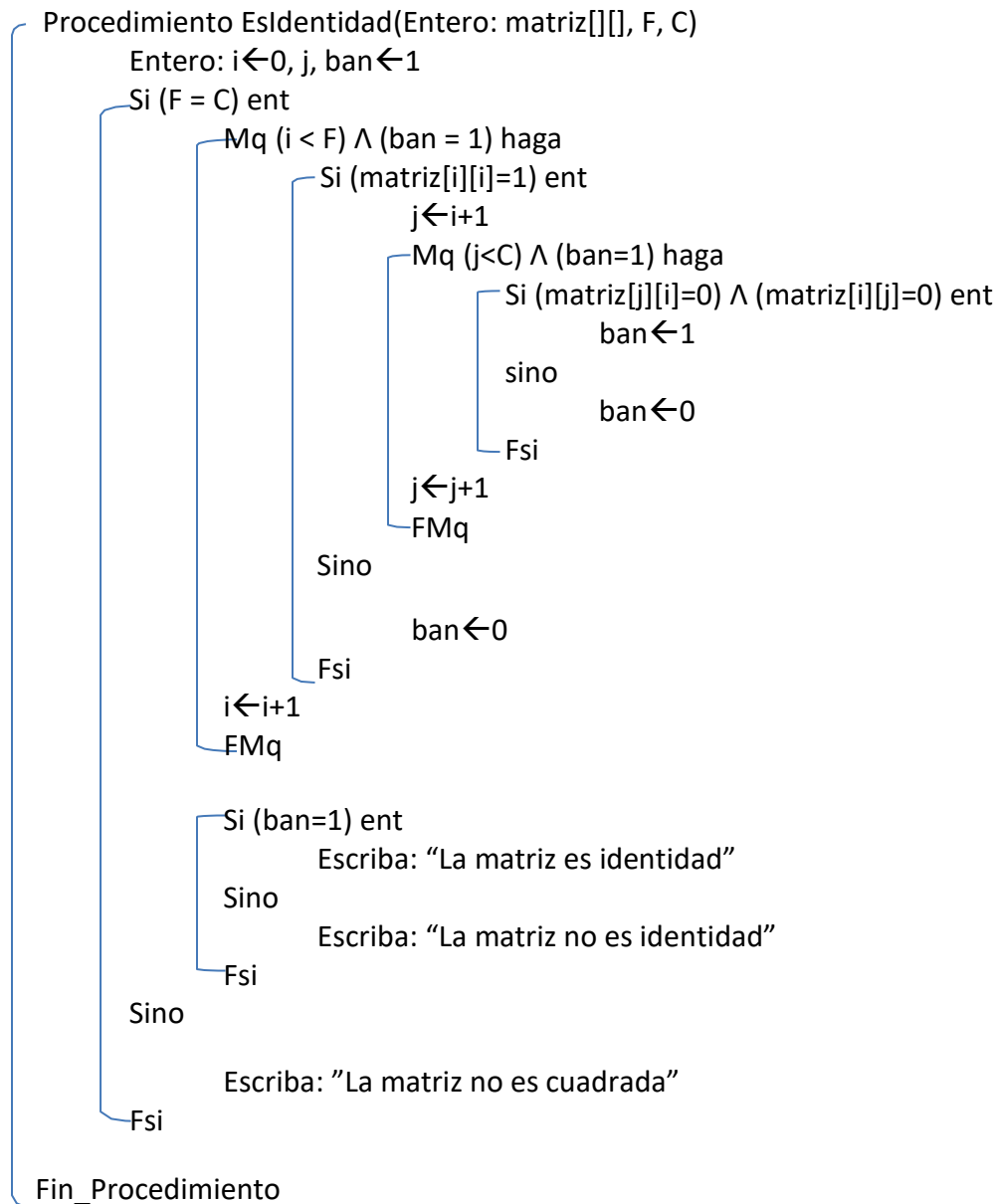
1	2	3
4	5	6
7	8	9





3. **MATRIZ IDENTIDAD:** Realice un subprograma que compruebe si una matriz es Identidad.

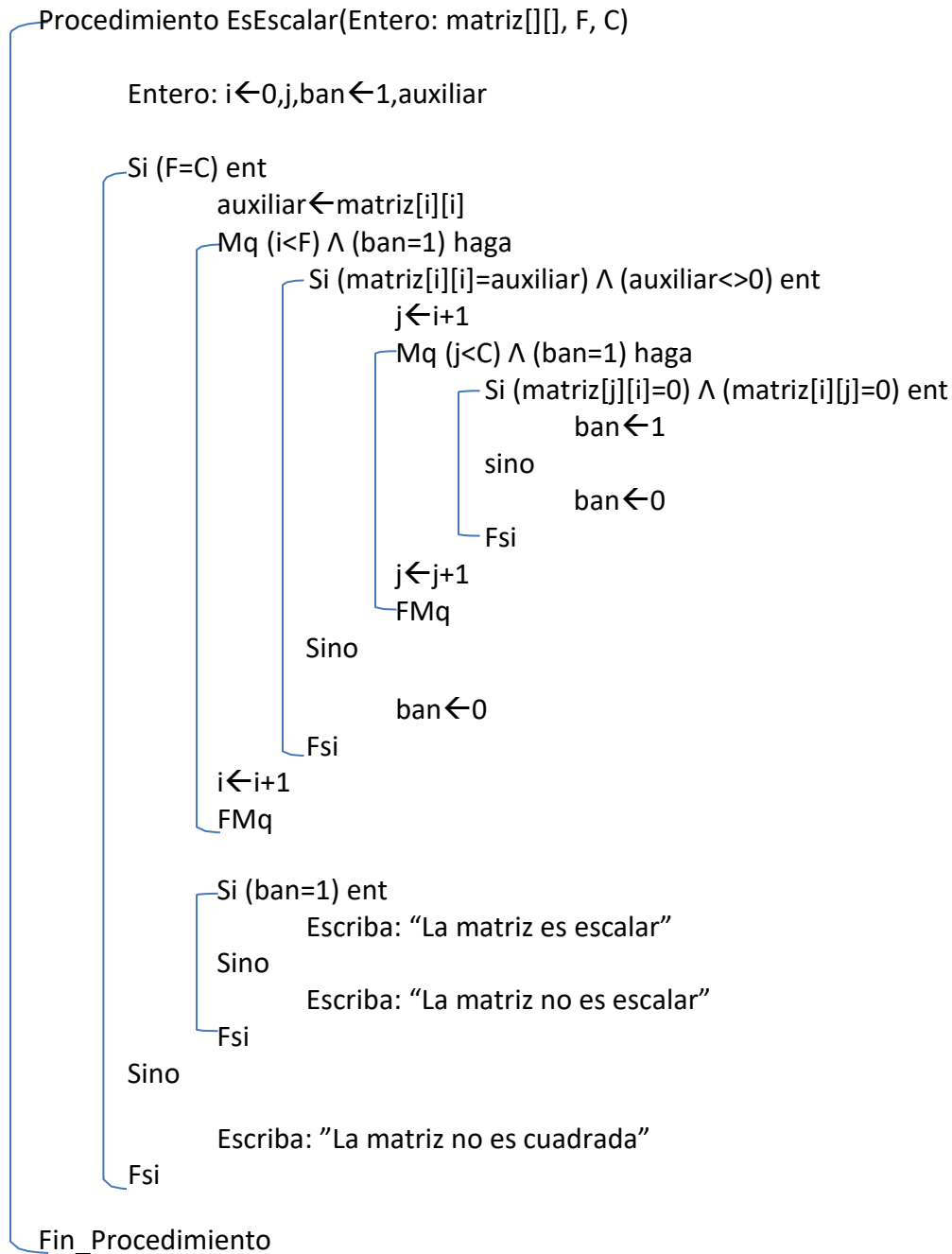
1	0	0
0	1	0
0	0	1





4. **MATRIZ ESCALAR:** Realice un subprograma que compruebe si una matriz es Escalar.

8	0	0
0	8	0
0	0	8





5. **MATRIZ TRANSPUESTA:** Realice un subprograma que compruebe si una matriz es la Transpuesta de otra.

A=

2	3	0
1	2	0
3	5	6

A^T=

2	1	3
3	2	5
0	0	6

Procedimiento Transpuesta (Entero: matriz[],matrizt[], F, C, Ft, Ct)

Entero: : i←0,j,ban←1

Si (F=Ct) ∧ (C=Ft) ent

Mq (i<F) ∧ (ban=1) haga

j←0

Mq (j<C) ∧ (ban=1) haga

Si (matriz[i][j]=matrizt[j][i]) ent

ban←1

sino

ban←0

Fsi

j←j+1

FMq

i←i+1

FMq

Sino

ban←0

Fsi

Si (ban=1) ent

Escriba: "La matriz si es la Transpuesta de matrizt"

Sino

Escriba: "La matriz no es la Transpuesta de matrizt"

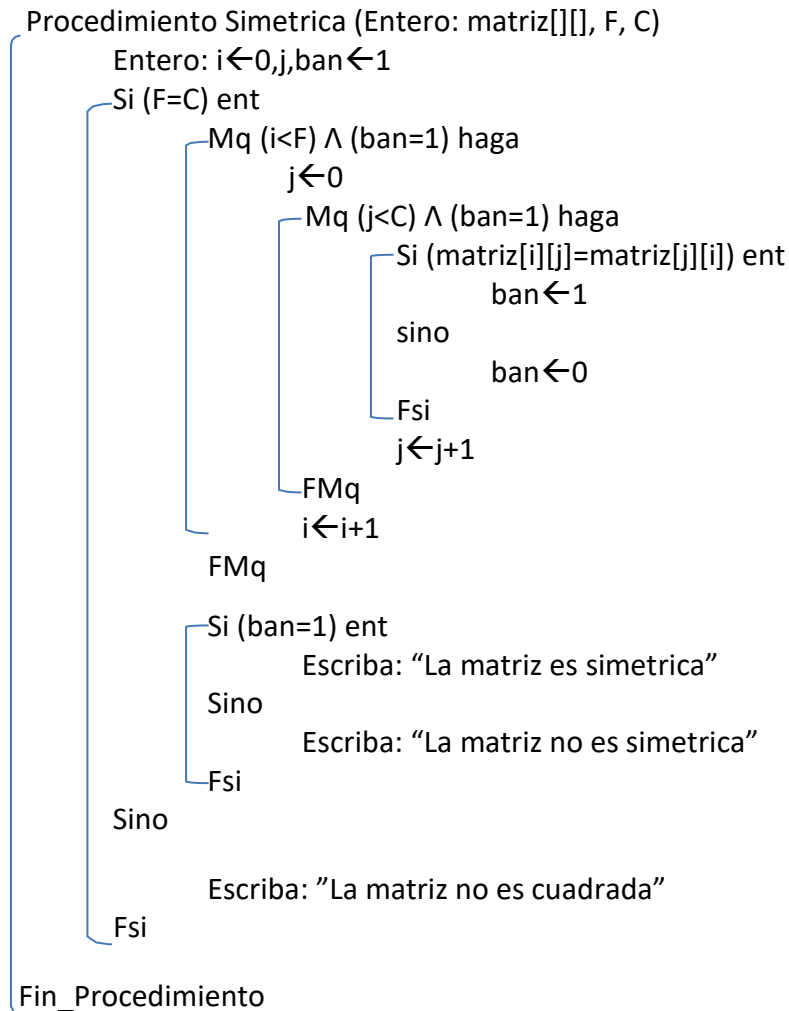
Fsi

Fin_Procedimiento



6. **MATRIZ SIMÉTRICA:** Realice un subprograma que compruebe si una matriz es simétrica.

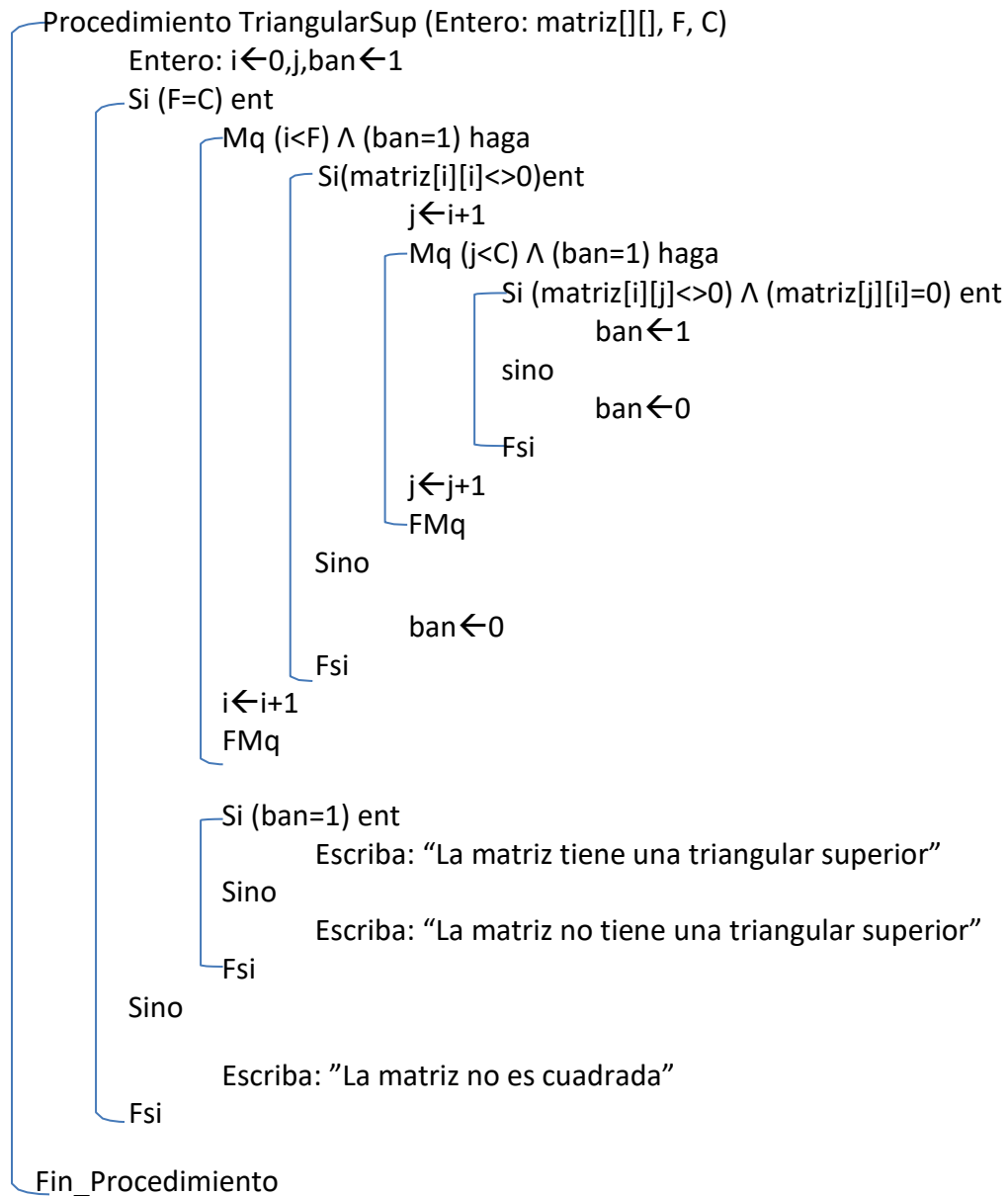
1	-1	3
-1	2	4
3	4	7





7. **TRIANGULAR SUPERIOR DE UNA MATRIZ:** Realice un subprograma que compruebe si una matriz tiene una triangular superior.

1	3	-2
0	4	8
0	0	5





8. **TRIANGULAR INFERIOR DE UNA MATRIZ:** Realice un subprograma que compruebe si una matriz tiene una triangular inferior.

1	0	0
3	5	0
-8	4	9

