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

```
Procedimiento DiagonalPrincipal (Entero: matriz[][], F, C)
        Entero: i
        Si (F=C) ent
                Para (i←0, F-1, 1)haga
                        Escriba: matriz[i][i]
                Fpara
        sino
                Escriba: ”La matriz no es cuadrada”
        Fsi

Fin_Procedimiento
```

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

```
Procedimiento DiagonalSecundaria (Entero: matriz[][], F, C)
        Entero: i, j
        Si (F=C) ent
                j←C
                Para (i←0, F-1, 1) haga
                        j←j-1
                        Escriba: matriz[i][j]
                Fpara
        Sino
                Escriba: ”La matriz no es cuadrada”
        Fsi

Fin_Procedimiento
```

3. <u>MATRIZ IDENTIDAD:</u> Realice un subprograma que compruebe si una matriz es Identidad.

1	**0**	**0**
0	**1**	**0**
0	**0**	**1**

```
Procedimiento EsIdentidad(Entero: matriz[][], F, C)
        Entero: i←0, j, ban←1
        Si (F = C) ent
                Mq (i < F) ∧ (ban = 1) haga
                        Si (matriz[i][i]=1) ent
                                j←i+1
                                Mq (j<C) ∧ (ban=1) haga
                                        Si (matriz[j][i]=0) ∧ (matriz[i][j]=0) ent
                                                ban←1
                                        sino
                                                ban←0
                                        Fsi
                                j←j+1
                                FMq
                        Sino

                                ban←0
                        Fsi
                i←i+1
                FMq

                Si (ban=1) ent
                        Escriba: "La matriz es identidad"
                Sino
                        Escriba: "La matriz no es identidad"
                Fsi
        Sino

                Escriba: "La matriz no es cuadrada"
        Fsi

Fin_Procedimiento
```

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

8	**0**	**0**
0	**8**	**0**
0	**0**	**8**

```
Procedimiento EsEscalar(Entero: matriz[][], F, C)

        Entero: i←0,j,ban←1,auxiliar

        Si (F=C) ent
                auxiliar←matriz[i][i]
                Mq (i<F) ∧ (ban=1) haga
                        Si (matriz[i][i]=auxiliar) ∧ (auxiliar<>0) ent
                                j←i+1
                                Mq (j<C) ∧ (ban=1) haga
                                        Si (matriz[j][i]=0) ∧ (matriz[i][j]=0) ent
                                                ban←1
                                        sino
                                                ban←0
                                        Fsi
                                j←j+1
                                FMq
                        Sino

                                ban←0
                        Fsi
                i←i+1
                FMq

                Si (ban=1) ent
                        Escriba: "La matriz es escalar"
                Sino
                        Escriba: "La matriz no es escalar"
                Fsi
        Sino

                Escriba: "La matriz no es cuadrada"
        Fsi

Fin_Procedimiento
```

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. <u>MATRIZ SIMÉTRICA:</u> Realice un subprograma que compruebe si una matriz es simétrica.

1	**-1**	**3**
-1	**2**	**4**
3	**4**	**7**

```
Procedimiento Simetrica (Entero: matriz[][], F, C)
        Entero: i←0,j,ban←1
        Si (F=C) ent
                Mq (i<F) Λ (ban=1) haga
                        j←0
                        Mq (j<C) Λ (ban=1) haga
                                Si (matriz[i][j]=matriz[j][i]) ent
                                        ban←1
                                sino
                                        ban←0
                                Fsi
                                j←j+1
                        FMq
                        i←i+1
                FMq

                Si (ban=1) ent
                        Escriba: "La matriz es simetrica"
                Sino
                        Escriba: "La matriz no es simetrica"
                Fsi
        Sino

                Escriba: "La matriz no es cuadrada"
        Fsi

Fin_Procedimiento
```

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**

```
Procedimiento TriangularSup (Entero: matriz[][], F, C)
        Entero: i←0,j,ban←1
        Si (F=C) ent
                Mq (i<F) ∧ (ban=1) haga
                        Si(matriz[i][i]<>0)ent
                                j←i+1
                                Mq (j<C) ∧ (ban=1) haga
                                        Si (matriz[i][j]<>0) ∧ (matriz[j][i]=0) ent
                                                ban←1
                                        sino
                                                ban←0
                                        Fsi
                                j←j+1
                                FMq
                        Sino

                                ban←0
                        Fsi
                i←i+1
                FMq

                Si (ban=1) ent
                        Escriba: "La matriz tiene una triangular superior"
                Sino
                        Escriba: "La matriz no tiene una triangular superior"
                Fsi
        Sino

                Escriba: "La matriz no es cuadrada"
        Fsi

Fin_Procedimiento
```

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**

```
Procedimiento TriangularInf (Entero: matriz[][], F, C)
        Entero: i←0,j,ban←1
        Si (F=C) ent
                Mq (i<F) ∧ (ban=1) haga
                        Si (matriz[i][i]<>0) ent
                                j←i+1
                                Mq (j<C) ∧ (ban=1) haga
                                        Si (matriz[j][i]<>0) ∧ (matriz[i][j]=0) ent
                                                ban←1
                                        sino
                                                ban←0
                                        Fsi
                                j←j+1
                                FMq
                        Sino

                                ban←0
                        Fsi
                i←i+1
                FMq

                Si (ban=1) ent
                        Escriba: "La matriz tiene una triangular inferior"
                Sino
                        Escriba: "La matriz no tiene una triangular inferior"
                Fsi
        Sino

                Escriba: "La matriz no es cuadrada"
        Fsi

Fin_Procedimiento
```