

Apellidos,
Nombre

DNI

La puntuación total del examen es 10 puntos. Para aprobar será necesario un mínimo de 5 puntos.

1. Considera el siguiente tipo para representar matrices de tamaño menor o igual a 30x30:

CONST

TAM = 30;

TYPE

Rango = 1 .. TAM;

Matriz = **RECORD**

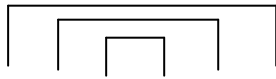
 NFilas, NCols : Rango;

 M : **ARRAY** [Rango, Rango] **OF** REAL

END;

Donde los dos primeros campos indican el tamaño de la matriz representada, y el tercero almacena los elementos de la matriz.

- a) (1.5 *puntos*) Define una función que tome una *Matriz* como parámetro y devuelva la que se obtiene al reflejar verticalmente dicha *Matriz*. Para ello, se intercambiarán la primera y última columna, la segunda y la penúltima, etc. Por ejemplo:



$$\begin{pmatrix} 1 & 7 & 9 & 8 & 10 & 17 \\ 13 & 2 & 4 & 25 & 6 & 22 \\ 4 & 0 & 5 & 12 & 15 & 20 \end{pmatrix}$$

matriz original

$$\begin{pmatrix} 17 & 10 & 8 & 9 & 7 & 1 \\ 22 & 6 & 25 & 4 & 2 & 13 \\ 20 & 15 & 12 & 5 & 0 & 4 \end{pmatrix}$$

matriz reflejada

- b) (2.5 *puntos*) Una matriz cuadrada es triangular inferior si todos los elementos por encima de la diagonal principal son nulos y los demás no lo son. Por ejemplo, la siguiente matriz es triangular superior:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 3 & 4 & 0 & 0 \\ 2 & 9 & 5 & 0 \\ 7 & 6 & 8 & 3 \end{pmatrix}$$

Escribe una función que tome como parámetro un *Matriz* y compruebe si es cuadrada y triangular superior o no. La función **debe ser eficiente**, en el sentido de que, si la matriz no fuese triangular superior, tan pronto como se pueda determinar esto se dejen de examinar los demás elementos de la matriz.

2. Se desea poder trabajar con números naturales de hasta 250 cifras, y dado que el tipo `CARDINAL` no permite representar números tan grandes, se opta por representar éstos como cadenas de caracteres con los dígitos de la representación decimal del número:

```

CONST
    TAM = 250;
TYPE
    Numero = String[TAM];

```

Por ejemplo, el número 1234 quedaría representado por la cadena '1234'.

En los siguientes apartados, supón que todos los caracteres que forman parte de las cadenas son dígitos, es decir que no se pasan como parámetros cadenas de caracteres que no representen números. Además, tampoco consideres la solución de convertir las cadenas a valores reales para operarlos. Define las funciones auxiliares que estimes necesarias.

- a) (2 *puntos*) Define una función `Por` que tome como parámetro una cadena de dígitos y **un único** carácter correspondiente a un dígito, y devuelva una cadena que represente el producto de ambos. Por ejemplo:

```
Por('329', '7') ==> '2303'
```

Para realizar el programa, recuerda como se realiza el producto “a mano”:

```

      2 6
      3 2 9
    x   7
    -----
      2 3 0 3

```

- b) (2 *puntos*) Usando la función anterior, define una función `Producto` que tome como parámetro dos cadenas de dígitos y devuelva una cadena que represente el producto de ambas. Por ejemplo:

```
Producto('12346', '123') ==> '1518558'
```

Para realizar el programa, recuerda como se realiza el producto “a mano”:

```

      1 2 3 4 6
    x   1 2 3
    -----
      3 7 0 3 8
      2 4 6 9 2 0
    + 1 2 3 4 6 0 0
    -----
      1 5 1 8 5 5 8

```

- c) (2 *puntos*) Define una función `Factorial` que tome como parámetro una cadena de dígitos y devuelva una cadena que represente su factorial. Por ejemplo:

```
Factorial('10') ==> '3628800'
```