

**Fundamentos de la programación y la informática**  
**Examen de ejercicios de laboratorios. 22 de enero de 2026**  
URJC. Grados en ingeniería aeroespacial. Turno de tarde

---

- Recuerda que si el programa no compila, su nota será nula.
- Puedes descargar el fichero donde quieras, pero si no tienes claro en qué carpeta trabajas, podrás hacer mal la entrega y suspender por eso.
- Cuando acabes, sube solamente el código fuente (fichero .pas) al aula virtual, apartado *evaluación*. No le cambies el nombre.

## Ejercicio único (10 puntos)

### Definiciones previas

Sea  $M = (m_{ij})$  una matriz bidimensional de  $M$  filas y  $N$  columnas, con  $M \geq 3$  y  $N \geq 3$ , cuyos índices cumplen  $1 \leq i \leq M$  y  $1 \leq j \leq N$ .

1. Definimos el *borde* de la matriz como el conjunto de elementos  $m_{ij}$  tales que  $i = 1$  o  $i = M$  o  $j = 1$  o  $j = N$ , y el *interior* como el conjunto de elementos  $m_{ij}$  tales que  $1 < i < M$  y  $1 < j < N$ .

Esto es: el *borde* de la matriz está formado por la primera fila, la última fila, la primera columna y la última columna. Y el interior, por los elementos que no están en el *borde*.

2. Denotamos por  $B$  el conjunto de índices correspondientes al borde y por  $I$  el conjunto de índices correspondientes al interior.
3. Dada la matriz  $M$ , definimos la *matriz de desviaciones*  $D = (d_{ij})$ , de las mismas dimensiones que  $M$ , como aquella cuyos elementos vienen dados por:

$$d_{ij} = \begin{cases} m_{ij} - E[B], & \text{si } (i, j) \in B, \\ m_{ij} - E[I], & \text{si } (i, j) \in I, \end{cases}$$

donde  $E[B]$  y  $E[I]$  representan la media aritmética de los elementos del borde y del interior, respectivamente, definida de la manera habitual.

Esto es: para calcular la matriz de desviaciones, tomamos cada elemento de la matriz original y le restamos la media de su zona (borde o interior). El resultado lo colocamos en la misma posición del elemento original.

### Especificación

Escribe un programa en Pascal que cumpla la siguiente especificación:

1. El nombre del fichero con el código fuente será exactamente *bordes.pas*, todo en minúsculas.
2. Estará escrito a partir de un esqueleto que encontrarás en el aula virtual, en *evaluación*, *bordes.pas*. Usa todo el esqueleto, añade lo que necesites pero no deberías modificar nada.
3. El programa tendrá un procedimiento llamado *genera\_informe\_matriz* que recibirá como argumento una matriz y:
  - a) La escribirá en pantalla.
  - b) Escribirá en pantalla el número de elementos del borde.

- c) Escribirá en pantalla el número de elementos del interior.
- d) Escribirá en pantalla la media aritmética de los elementos del borde.
- e) Escribirá en pantalla la media aritmética de los elementos del interior.
- f) Escribirá en pantalla la matriz de desviaciones.

Naturalmente, este procedimiento llamará a todos los subprogramas que consideres oportuno. Usa preferentemente funciones excepto cuando claramente lo adecuado sea escribir un procedimiento.

4. Para que puedas comprobar fácilmente que tu programa funciona bien, en el esqueleto proporcionado tienes una función que devuelve una matriz aleatoria y también una función que devuelve siempre una matriz *test* que siempre es la misma.

El cuerpo del programa principal:

- a) Comprobará que las matrices tengan al menos 3 filas y al menos 3 columnas, tal y como exige la definición previa. En caso contrario, mostrará un mensaje de error y concluirá.
- b) Llamará a *genera\_informe\_matriz* pasándole la matriz de prueba. Si tu código es correcto, el resultado será igual que el que verás a continuación.
- c) Luego llamará a *genera\_informe\_matriz* pasándole la matriz aleatoria. Naturalmente, estos resultados irán cambiando.

La salida del programa debería ser similar a esta:

```
Matriz test
41.89 32.32 23.75 14.18 5.61 96.04
78.06 69.49 60.92 51.35 42.78 33.21
15.23 6.66 97.09 88.52 79.95 70.38
52.40 43.83 34.26 25.69 16.12 7.55
```

```
Elementos del borde:16
Elementos del interior:8
Media del borde:36.91
Media del interior:62.09
Matriz de desviaciones:
4.98 -4.59 -13.16 -22.73 -31.30 59.13
41.15 7.39 -1.17 -10.74 -19.31 -3.70
-21.68 -55.43 34.99 26.42 17.85 33.47
15.49 6.92 -2.65 -11.22 -20.79 -29.36
```

```
Matriz aleatoria
0.01 97.38 23.65 43.99 44.11 8.97
43.30 0.78 57.79 37.80 39.71 75.30
70.92 90.81 84.72 30.62 77.99 82.26
99.95 56.86 62.55 35.55 90.25 61.11
```

```
Elementos del borde:16
Elementos del interior:8
Media del borde:56.01
Media del interior:52.53
Matriz de desviaciones:
-56.00 41.37 -32.36 -12.02 -11.90 -47.04
-12.71 -51.75 5.26 -14.73 -12.82 19.29
14.91 38.28 32.19 -21.91 25.46 26.25
43.94 0.85 6.54 -20.46 34.24 5.10
```

## Observaciones

- Es imprescindible que escribas las matrices en pantalla usando el procedimiento del esqueleto.
- Para calcular cuántos elementos tiene el borde / el interior, puedes usar un método iterativo o un método analítico, como prefieras.
- Obviamente, este programa tiene que seguir funcionando si modificamos el valor de las constantes *Filas* y *Columnas*, no es válido que escribas un programa que solo funcione para 4 filas y 6 columnas.

## Esqueleto proporcionado

<https://gsyc.urjc.es/~mortuno/bordes.pas>

## Solución

```
{ $mode objfpc } { $H- } { $R+ } { $T+ } { $Q+ } { $V+ } { $D+ } { $X- } { $warnings on }
program bordes;
// Nombre:
// Apellidos:
const
    Filas = 4;
    Columnas = 6;
    CotaSuperior = 100;
type
    TipoMatriz = array[1..Filas, 1..Columnas] of real;

function genera_numero(cota_superior:integer):real;
    // Genera un número aleatorio entre 0 y cota_superior,
    // con 2 decimales
begin
    result := random(cota_superior) + random(100) / 100.0;
end;

function genera_matriz_random():TipoMatriz;
    // Esta FUNCION, sin argumentos, genera una matriz aleatoria.
var
    i,j : integer;
begin
    for i := 1 to Filas do begin
        for j:= 1 to Columnas do
            result[i,j] := genera_numero(CotaSuperior);
        end;
    end;
end;

function genera_matriz_test():TipoMatriz;
    // Esta FUNCION, sin argumentos, siempre genera la misma matriz,
    // que usaremos como prueba.
var
    i, j: integer;
    parte_entera: integer;
    parte_decimal: integer;
begin
    for i := 1 to Filas do begin
        for j := 1 to Columnas do begin
            parte_entera := (i * 37 + j * 91 + 13) mod CotaSuperior;

            // Decimal entre 0 y 99 -> .00 a .99
            parte_decimal := (i * 17 + j * 43 + 29) mod 100;

            result[i,j] := parte_entera + (parte_decimal / 100.0);
        end;
    end;
```

```

        end;
end;

procedure escribe_matriz(matriz:TipoMatriz);
var i,j: integer;
begin
    for i := 1 to Filas do begin
        for j:= 1 to Columnas do begin
            write(matriz[i,j]:7:2);
        end;
        writeln;
    end;
    writeln;
end;

// ----- Aquí acaba el esqueleto.

function es_borde(i, j: integer): boolean;
begin
    result := (i = 1) or (i = Filas) or (j = 1) or (j = Columnas);
end;

function cuenta_elementos_borde(matriz: TipoMatriz): integer;
// Realmente el parámetro 'matriz' no es necesario, pero
// mantenerlo posiblemente resulta más claro.
var
    i, j: integer;
begin
    result := 0;
    for i := 1 to Filas do begin
        for j := 1 to Columnas do begin
            if es_borde(i, j) then
                result := result +1;
            end;
        end;
    end;
end;

function cuenta_elementos_interior(matriz: TipoMatriz): integer;
// Realmente el parámetro 'matriz' no es necesario, pero
// mantenerlo posiblemente resulta más claro.
var
    i, j: integer;
begin
    result := 0;
    for i := 1 to Filas do begin
        for j := 1 to Columnas do begin
            if not es_borde(i, j) then
                result := result +1;
            end;
        end;
    end;
end;
// También podríamos calcular el número de elementos de forma analítica:
// En el borde hay 2 * Filas + 2 * Columnas - 4

```

```

// En el interior, (Filas * Columnas) - Elementos del borde

function calcula_media_borde(matriz: TipoMatriz; elementos_borde: integer):
    real;
var
    i, j: integer;
    suma: real;
begin
    suma := 0.0;

    for i := 1 to Filas do begin
        for j := 1 to Columnas do begin
            if es_borde(i, j) then begin
                suma := suma + matriz[i,j];
            end;
        end;
    end;

    result := suma / elementos_borde;
end;

function calcula_media_interior(matriz: TipoMatriz; elementos_interior:
    integer): real;
var
    i, j: integer;
    suma: real;
begin
    suma := 0.0;

    for i := 1 to Filas do begin
        for j := 1 to Columnas do begin
            if not es_borde(i, j) then begin
                suma := suma + matriz[i,j];
            end;
        end;
    end;

    result := suma / elementos_interior;
end;

```

```

function calcula_matriz_desv(matriz: TipoMatriz;
    media_borde, media_interior: real): TipoMatriz;
var
    i, j: integer;
begin
    for i := 1 to Filas do begin
        for j := 1 to Columnas do begin
            if es_borde(i, j) then
                result[i,j] := matriz[i,j] - media_borde
            else
                result[i,j] := matriz[i,j] - media_interior;
            end;
        end;
    end;
end;

procedure genera_informe_matriz(matriz: TipoMatriz);
var
    media_borde, media_interior: real;
    matriz_desviaciones: TipoMatriz;
    elementos_borde, elementos_interior: integer;
begin
    elementos_borde:=cuenta_elementos_borde(matriz);
    elementos_interior:=cuenta_elementos_interior(matriz);
    media_borde:=calcula_media_borde(matriz, elementos_borde);
    media_interior:=calcula_media_interior(matriz, elementos_interior);
    matriz_desviaciones := calcula_matriz_desv(matriz, media_borde,
        media_interior);
    escribe_matriz(matriz);
    writeln('Elementos del borde:', elementos_borde);
    writeln('Elementos del interior:', elementos_interior);
    writeln('Media del borde:', media_borde:0:2);
    writeln('Media del interior:', media_interior:0:2);
    writeln('Matriz de desviaciones:');

    escribe_matriz(matriz_desviaciones);
end;

var
    matriz: TipoMatriz;

const
    DimensionMinima=3;

begin
    randomize();

    if (Filas < DimensionMinima) or (Columnas < DimensionMinima) then begin
        write('La matriz debe tener al menos ', DimensionMinima);
        writeln(' filas y ', DimensionMinima, ' columnas');
    end
    else begin
        writeln('Matriz test');
    end
end

```

```
matriz := genera_matriz_test();
genera_informe_matriz(matriz);

writeln('Matriz aleatoria');
matriz := genera_matriz_random();
genera_informe_matriz(matriz);
end;

end.
```