

PRÁCTICA 1

Resolución de ejercicios y preguntas propuestas acerca de la práctica 1.

→ Ejercicio 1

Enunciado

Repita estas operaciones con otra imagen 'bacteria.tif'. ¿Cuál es su tamaño? ¿Cuántos niveles de grises puede tener? ¿Cuál es el valor mínimo y el máximo?

Resolución

The screenshot shows the MATLAB Editor with a script named 'Ejercicio1.m'. The script performs the following steps:

- Clears the workspace and command window.
- Closes all figures except the command window.
- Resets the workspace.
- Loads the image 'bacteria.tif' into the variable 'imgBacteria'.
- Displays the size of 'imgBacteria'.
- Displays the class of 'imgBacteria'.
- Calculates the minimum and maximum values of 'imgBacteria' using 'min' and 'max' with the colon operator to convert the matrix into a single column vector.

The Workspace window on the right shows the following variables:

Name	Value
ans	'uint8'
imgBacteria	178x178 uint8
Max	239
Min	0

The Command Window shows the output of the 'Max' variable:

```
Max =
uint8
239
```

Transcripción de script

```
%cargamos la imagen

imgBacteria = imread('bacteria.tif');

imshow(imgBacteria)

% ¿Cuál es su tamaño?

size(imgBacteria)

% ¿Cuántos niveles de grises puede tener?

class(imgBacteria)

% ¿Cuál es el valor mínimo y máximo?

Min = min(imgBacteria(:)) % uso ':' para convertir la matriz en un solo vector columna

Max = max(imgBacteria(:))

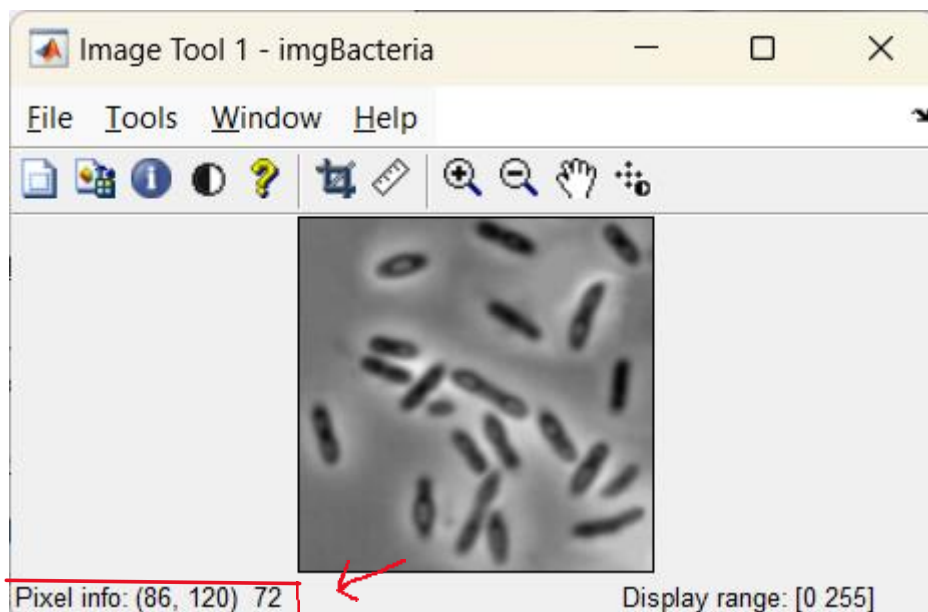
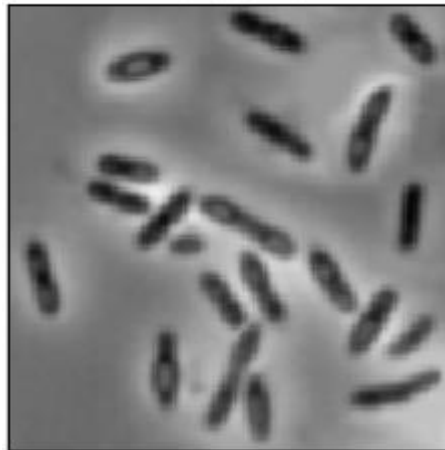
% examinar con imtool

imtool(imgBacteria)
```

Respuesta

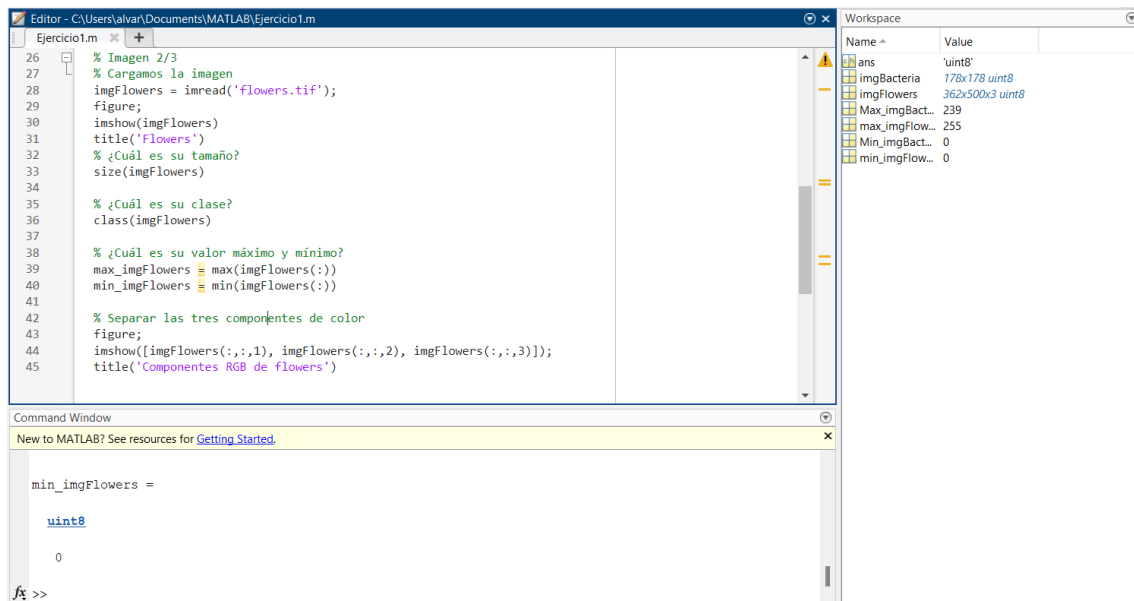
Es una imagen en **escala de grises** de **178x178 píxeles**, al ser de tipo **uint8** los valores pueden estar entre 0 y 255 (256 posibles niveles). Su valor **mínimo** es 0 y su valor **máximo** es 239.

Bacteria

**Enunciado**

Vuelva a realizar las mismas operaciones de: a) lectura, b) tamaño y clase de la imagen y c) visualización sobre una imagen de color 'flowers.tif'.

Resolución



Transcripción de script

```

% Imagen 2/3
% Cargamos la imagen
imgFlowers = imread('flowers.tif');
figure;
imshow(imgFlowers)
title('Flowers')
% ¿Cuál es su tamaño?
size(imgFlowers)

% ¿Cuál es su clase?
class(imgFlowers)

% ¿Cuál es su valor máximo y mínimo?
max_imgFlowers = max(imgFlowers(:))
min_imgFlowers = min(imgFlowers(:))

% Separar las tres componentes de color
figure;
imshow([imgFlowers(:, :, 1), imgFlowers(:, :, 2), imgFlowers(:, :, 3)]);
title('Componentes RGB de flowers')

% examinar con imtool
imtool(imgFlowers)
  
```

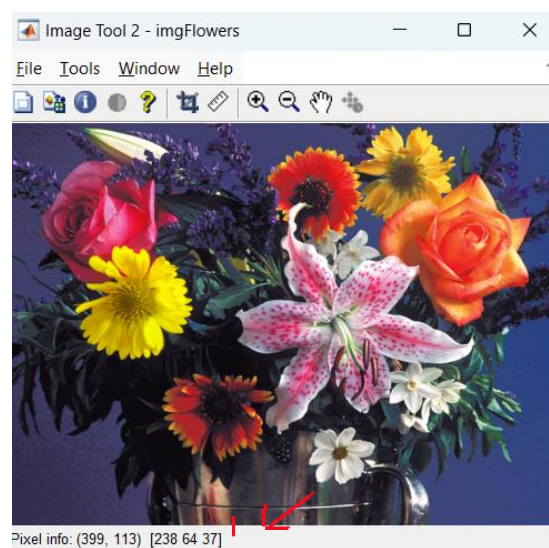
Respuesta

Se trata de una **imagen a color** de **362x500 píxeles** (RGB ya que cuenta con 3 canales), al ser de tipo **uint8** los valores pueden estar entre **0 y 255** (256 posibles niveles). Su valor **mínimo** es 0 y su valor **máximo** es 255.

Flowers



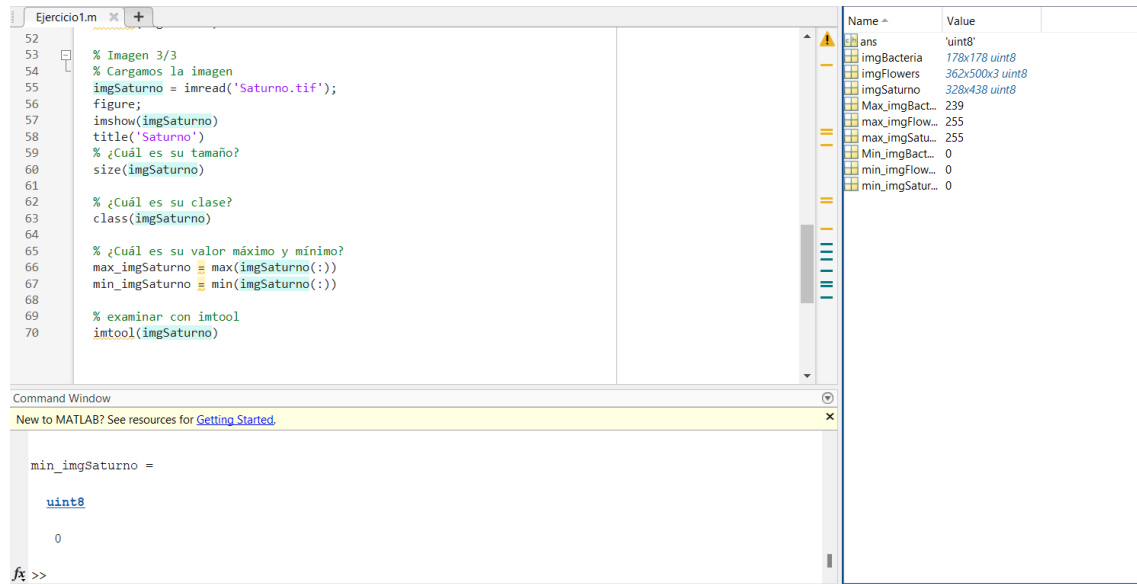
Componentes RGB de flowers



Enunciado

Vuelva a realizar las mismas operaciones de: a) lectura, b) tamaño y clase de la imagen y c) visualización sobre una imagen, en este caso hemos escogido 'Saturno.tif'.

Resolución



Transcripción de script

```

% Imagen 3/3
% Cargamos la imagen
imgSaturno = imread('Saturno.tif');
figure;
imshow(imgSaturno)
title('Saturno')
% ¿Cuál es su tamaño?
size(imgSaturno)

% ¿Cuál es su clase?
class(imgSaturno)

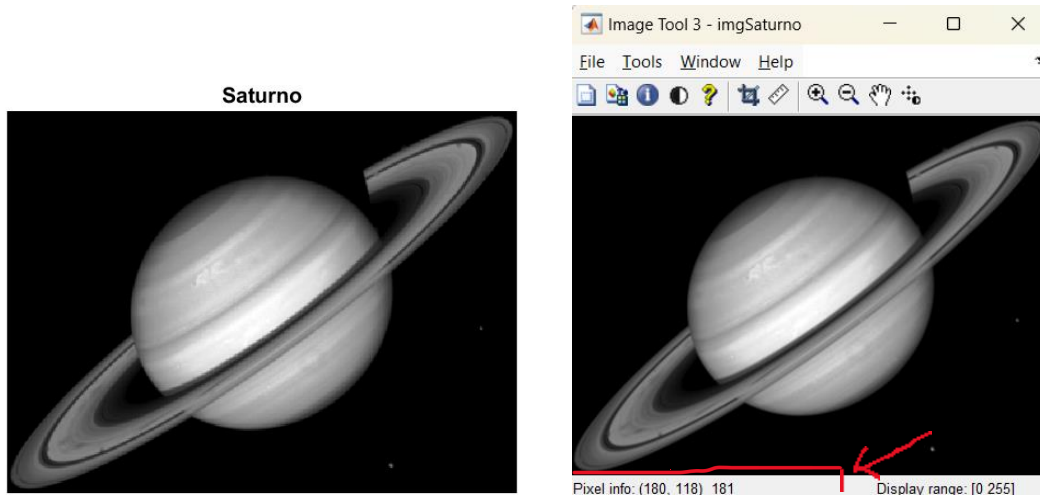
% ¿Cuál es su valor máximo y mínimo?
max_imgSaturno = max(imgSaturno(:))
min_imgSaturno = min(imgSaturno(:))

% examinar con imtool
imtool(imgSaturno)

```

Respuesta

Es una imagen en **escala de grises de 328x438 píxeles**, al ser de tipo **uint8** los valores pueden estar entre 0 y 255 (256 posibles niveles). Su valor **mínimo** es 0 y su valor **máximo** es 255.



→ Ejercicio 2

Enunciado

Realice la misma operación de binarización con la imagen 'coins.png'. Convierta a imagen indexada una imagen a color.

Imagen original (escala de grises)



Resolución

```

Editor - C:\User\alvar\Documents\MATLAB\Ejercicio1.m
Ejercicio1.m
73 % Ejercicio 2
74 imgEntGris = imread('coins.png');
75
76 % Mostrar imagen original
77 figure(1);
78 imshow(imgEntGris);
79 title('Imagen original (escala de grises)');
80 impixelinfo;
81
82 % Convertir imagen binaria
83 imgBW = im2bw(imgEntGris);
84
85 % Mostrar la imagen binaria
86 figure(2);
87 imshow(imgBW);
88 title('Imagen binaria');
89 impixelinfo;
90

```

Command Window

New to MATLAB? See resources for [Getting Started](#).

```

min_imgSaturno =
    uint8
    0
fx >>

```

Workspace

Name	Value
ans	'uint8'
imgBacteria	178x178 uint8
imgBW	246x300 logical
imgEntGris	246x300 uint8
imgFlowers	362x500x3 uint8
imgSaturno	328x438 uint8
Max_imgBact...	239
max_imgFlow...	255
max_imgSatu...	255
Min_imgBact...	0
min_imgFlow...	0
min_imgSatu...	0

Zoom: 100% UTF-8 CRLF script Ln 81 Col 1

Transcripción de script

```
% Ejercicio 2
imgEntGris = imread('coins.png');

% Mostrar imagen original
figure;
imshow(imgEntGris);
title('Imagen original (escala de grises)');
impixelinfo;

% Convertir imagen binaria
imgBW = im2bw(imgEntGris);

% Mostrar la imagen binaria
figure;
imshow(imgBW);
title('Imagen binaria');
impixelinfo;

% Convertir a imagen indexada una a color
% Utilizare de nuevo 'flowers'
I_color = imread('flowers.tif');

% Especificar el número de colores para la imagen indexada
numColors = 7;

% Convertir la imagen RGB a una imagen indexada
[indexedImage, cmap] = rgb2ind(I_color, numColors);

% Mostrar la imagen indexada con su mapa de colores
figure;
imshow(indexedImage, cmap);
title('Imagen indexada');
colormap(cmap);
colorbar;

% Valores de los píxeles
impixelinfo;
```

Respuesta



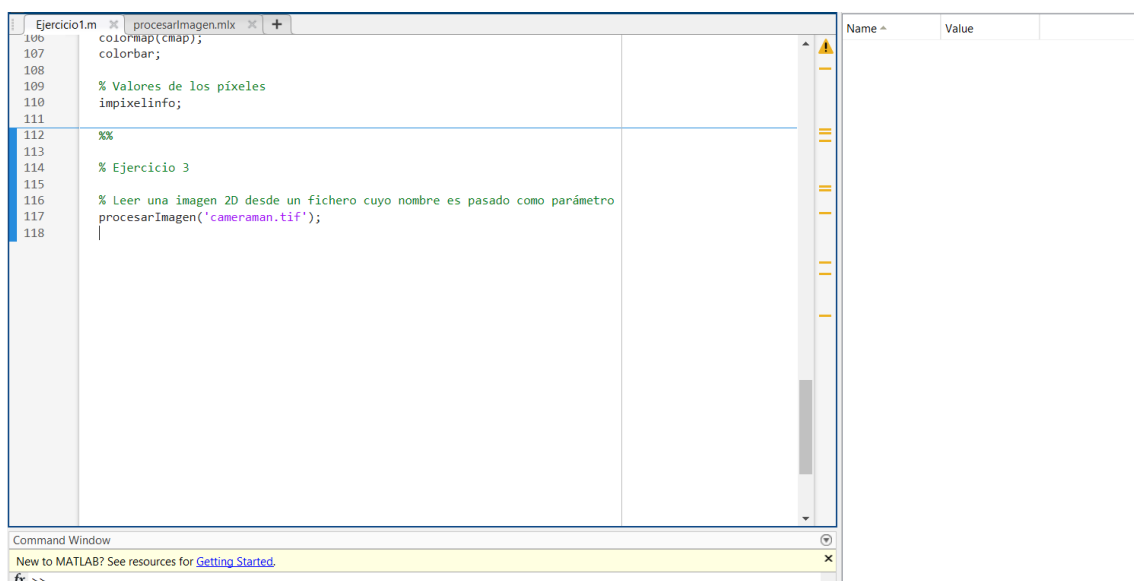
→ Ejercicio 3

Enunciado

En este apartado se tratará de realizar la primera función (*.m) de procesamiento de imágenes con Matlab. Consistirá en leer un fichero de imagen 2D, cuyo nombre es pasado por parámetro, se visualizará y se aplicará una umbralización automática, la cual es también visualizada.

Busca qué es un fichero .mlx (live script) y génalo con lo mismo que antes.

Resolución



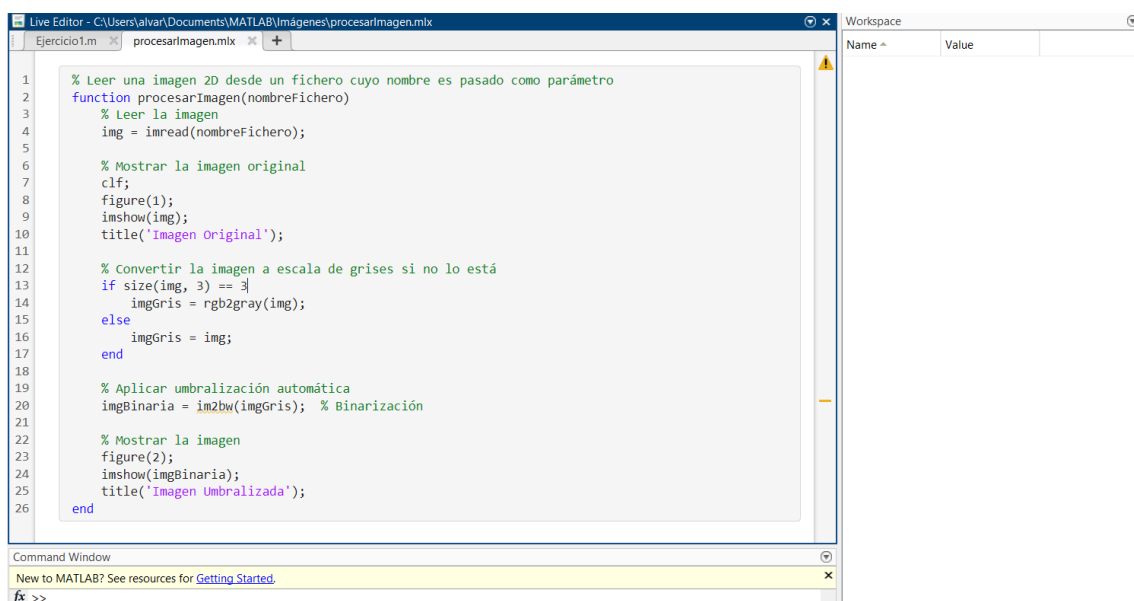
```

106 colormap(cmap);
107 colorbar;
108
109 % Valores de los píxeles
110 impixelinfo;
111
112 %%
113
114 % Ejercicio 3
115
116 % Leer una imagen 2D desde un fichero cuyo nombre es pasado como parámetro
117 procesarImagen('cameraman.tif');
118

```

Command Window

New to MATLAB? See resources for [Getting Started](#).



```

1 % Leer una imagen 2D desde un fichero cuyo nombre es pasado como parámetro
2 function procesarImagen(nombreFichero)
3 % Leer la imagen
4 img = imread(nombreFichero);
5
6 % Mostrar la imagen original
7 clf;
8 figure(1);
9 imshow(img);
10 title('Imagen Original');
11
12 % Convertir la imagen a escala de grises si no lo está
13 if size(img, 3) == 3
14     imgGris = rgb2gray(img);
15 else
16     imgGris = img;
17 end
18
19 % Aplicar umbralización automática
20 imgBinaria = im2bw(imgGris); % Binarización
21
22 % Mostrar la imagen
23 figure(2);
24 imshow(imgBinaria);
25 title('Imagen Umbralizada');
26 end

```

Command Window

New to MATLAB? See resources for [Getting Started](#).

Transcripción de script

```
% Leer una imagen 2D desde un fichero cuyo nombre es pasado como parámetro
function procesarImagen(nombreFichero)
% Leer la imagen
img = imread(nombreFichero);

% Mostrar la imagen original
clf;
figure(1);
imshow(img);
title('Imagen Original');

% Convertir la imagen a escala de grises si no lo está
if size(img, 3) == 3
    imgGris = rgb2gray(img);
else
    imgGris = img;
end

% Aplicar umbralización automática
imgBinaria = im2bw(imgGris); % Binarización

% Mostrar la imagen
figure(2);
imshow(imgBinaria);
title('Imagen Umbralizada');
end

% Ejercicio 3

% Leer una imagen 2D desde un fichero cuyo nombre es pasado como parámetro
procesarImagen('cameraman.tif');
```

Respuesta

Un fichero **.mlx** o **Live Script**, es un formato de archivo utilizado en MATLAB para crear **scripts interactivos**. Estos archivos permiten compartir resultados y código de manera interactiva y rápida.

Imagen Original



Imagen Umbralizada

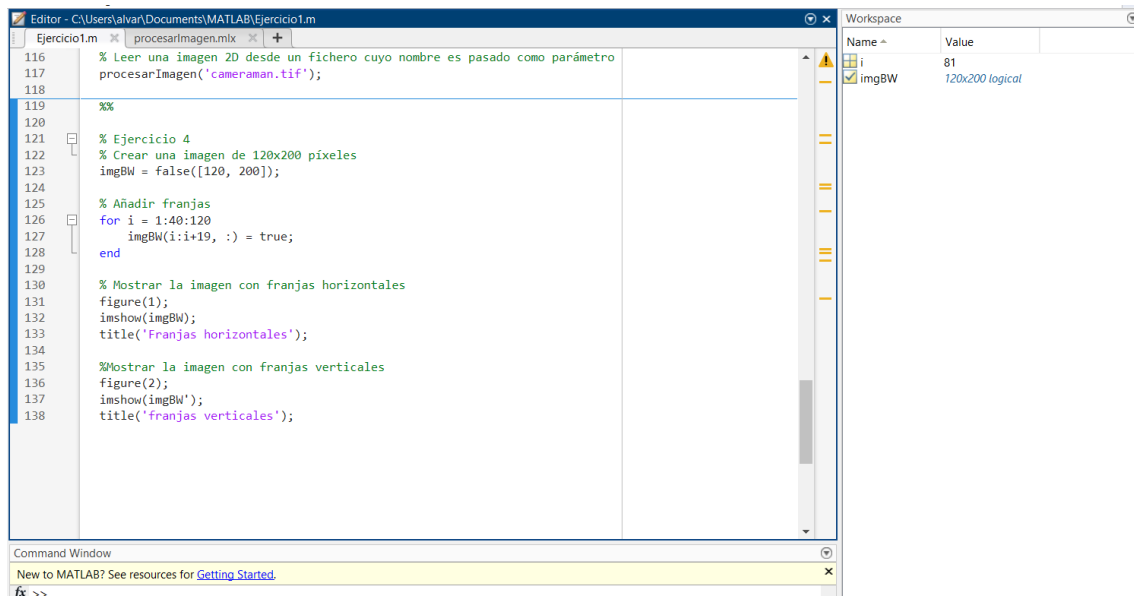


➔ Ejercicio 4

Enunciado

Construir imagen binaria de 120 x 200 píxeles que tenga franjas horizontales de 20 píxeles de anchura, distanciada por cada 20 píxeles. Si queremos que las franjas sean verticales sólo habría que emplear el operador traspuesta de las matrices.

Resolución



Transcripción de script

```

% Ejercicio 4
% Crear una imagen de 120x200 píxeles
imgBW = false([120, 200]);

% Añadir franjas
for i = 1:40:120
    imgBW(i:i+19, :) = true;
end

% Mostrar la imagen con franjas horizontales
figure(1);
imshow(imgBW);
title('Franjas horizontales');

%Mostrar la imagen con franjas verticales
figure(2);
imshow(imgBW');
title('franjas verticales');

```

Respuesta



→ Ejercicio 5

Enunciado

Realizar una función que construya y visualice dos imágenes de 256x256 con variación del nivel de gris en filas y columnas.

Resolución

```

Editor - C:\Users\alvar\Documents\MATLAB\Ejercicio1.m
Ejercicio1.m  procesarImagen.mlx  +
137 imshow(imgBW);
138 title('franjas verticales');
139
140 %%
141 % Ejercicio 5
142 function construirImagenes()
143     % Dimensiones
144     filas = 256;
145     columnas = 256;
146
147     % Crear una imagen con variación del nivel de gris en filas
148     imgFilas = repmat((0:255)', 1, columnas);
149
150     % Crear una imagen con variación del nivel de gris en las columnas
151     imgColumnas = repmat(0:255, filas, 1);
152
153     % Visualizar la imagen con variación en filas
154     figure(1);
155     imshow(uint8(imgFilas)); % Convierto a uint8 para mostrar
156     title('Franjas horizontales');
157
158     % Visualizar la imagen con variación en columnas
159     figure(2);
160     imshow(uint8(imgColumnas)); % Convierto a uint8 para mostrar
161     title('Franjas verticales');
162 end
163
164 construirImagenes()

```

Name	Value
ans	'uint8'
cmap	7x3 double
i	81
L_color	362x500x3 uint8
imgBacteria	178x178 uint8
imgBW	120x200 logical
imgEntGris	246x300 uint8
imgFlowers	362x500x3 uint8
imgSaturno	328x438 uint8
indexedImage	362x500 uint8
Max_imgBact...	239
max_imgFlow...	255
max_imgSatur...	255
Min_imgBact...	0
min_imgFlow...	0
min_imgSatur...	0
numColors	7

Command Window
New to MATLAB? See resources for [Getting Started](#).

Transcripción de script

```
% Ejercicio 5
function construirImagenes()
% Dimensiones
filas = 256;
columnas = 256;

% Crear una imagen con variación del nivel de gris en filas
imgFilas = repmat((0:255)', 1, columnas);

% Crear una imagen con variación del nivel de gris en las columnas
imgColumnas = repmat(0:255, filas, 1);

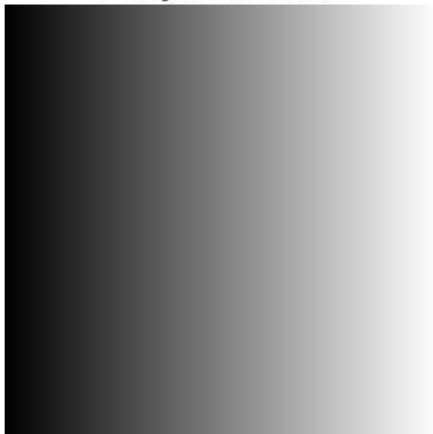
% Visualizar la imagen con variación en filas
figure(1);
imshow(uint8(imgFilas)); % Convierto a uint8 para mostrar
title('Franjas horizontales');

% Visualizar la imagen con variación en columnas
figure(2);
imshow(uint8(imgColumnas)); % Convierto a uint8 para mostrar
title('Franjas verticales');
end

construirImagenes()
```

Resultado

Franjas verticales



Franjas horizontales

