UNIVERSITY OF THE PUNJAB

Gujranwala Campus

DEPARTMENT OF INFORMATION TECHNOLOGY

Assignment #01

Course Title:

" COMPUTER VISION"

Submitted by:

Farheen Akhtar

BIT21029

BS-IT(7th Semester-Morning)

Submitted to:

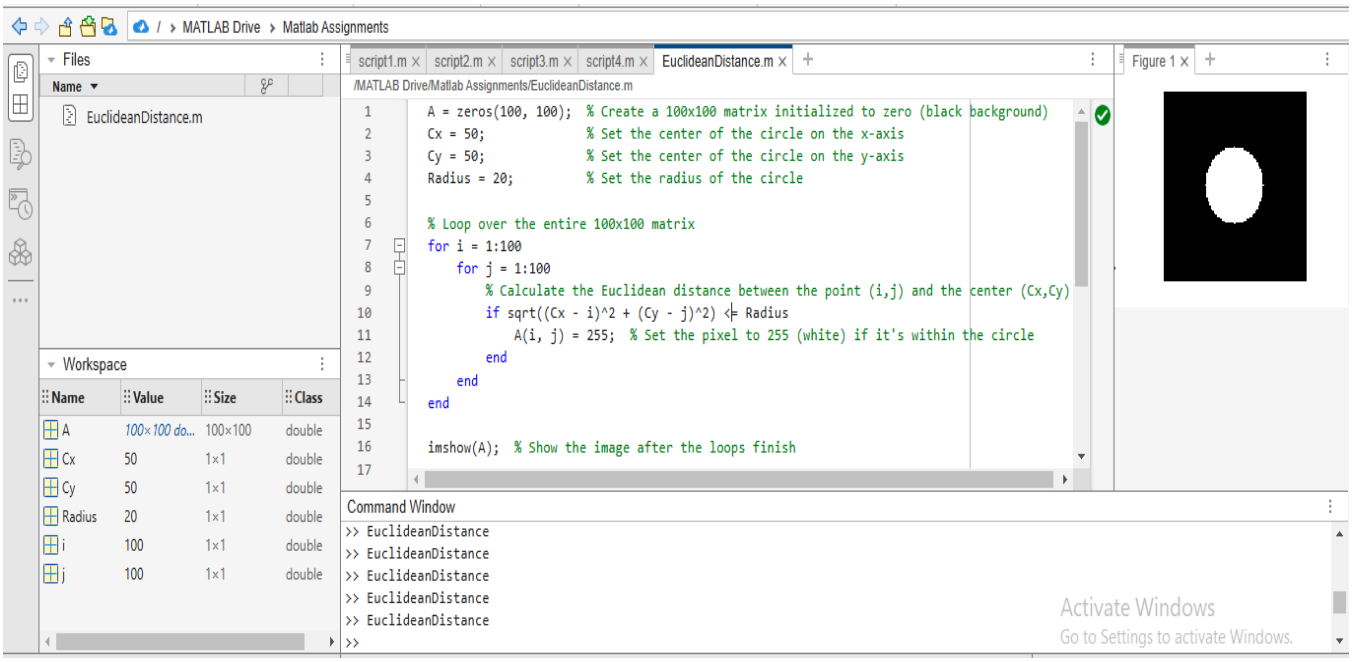
Mam Fouqia Zafeer

Dated: Jan 17,2025

Topic:

“Measuring Distance”

1. Euclidean Distance:



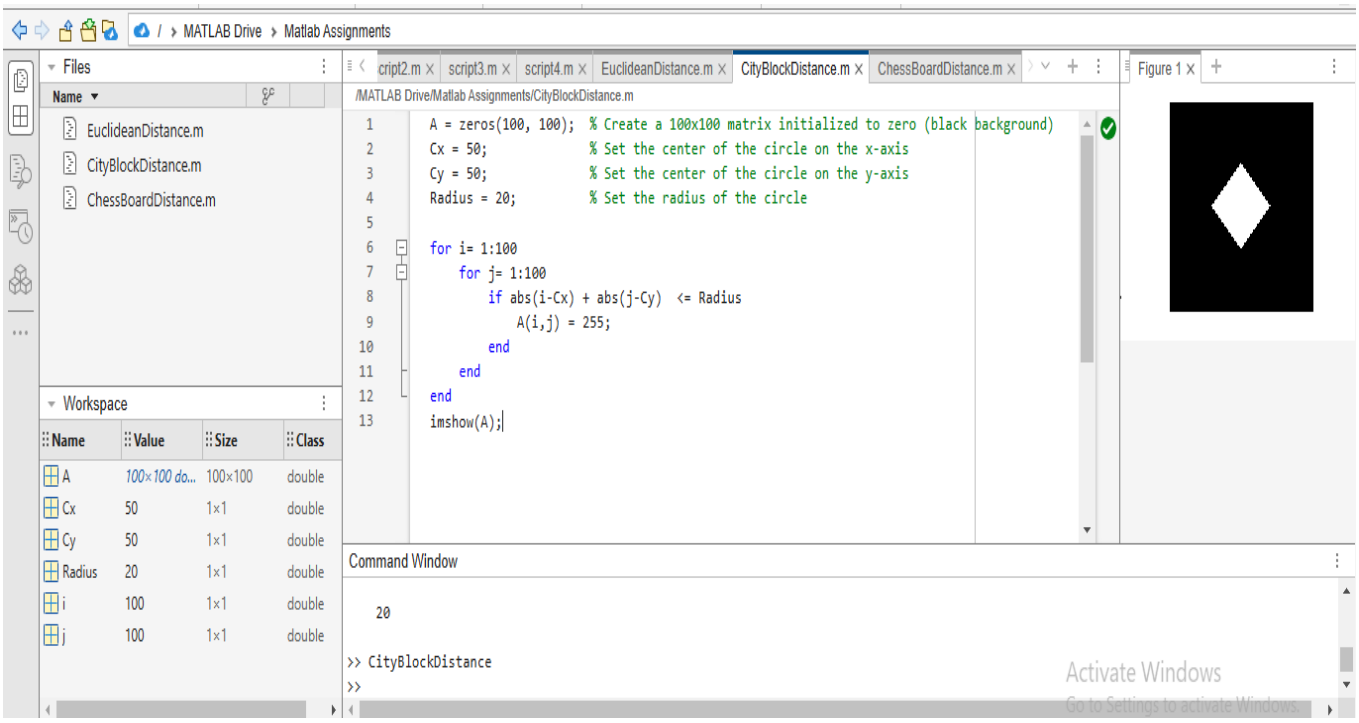
Matlab Code:

```
A = zeros(100, 100);  % Create a 100x100 matrix initialized to zero (black background)
Cx = 50;              % Set the center of the circle on the x-axis
Cy = 50;              % Set the center of the circle on the y-axis
Radius = 20;          % Set the radius of the circle

% Loop over the entire 100x100 matrix
for i = 1:100
   for j = 1:100
      % Calculate the Euclidean distance between the point (i,j) and the center (Cx,Cy)
      if sqrt((Cx - i)^2 + (Cy - j)^2) <= Radius
         A(i, j) = 255;  % Set the pixel to 255 (white) if it's within the circle
      end
   end
end

imshow(A);  % Show the image after the loops finish
```

2. City-Block Distance:

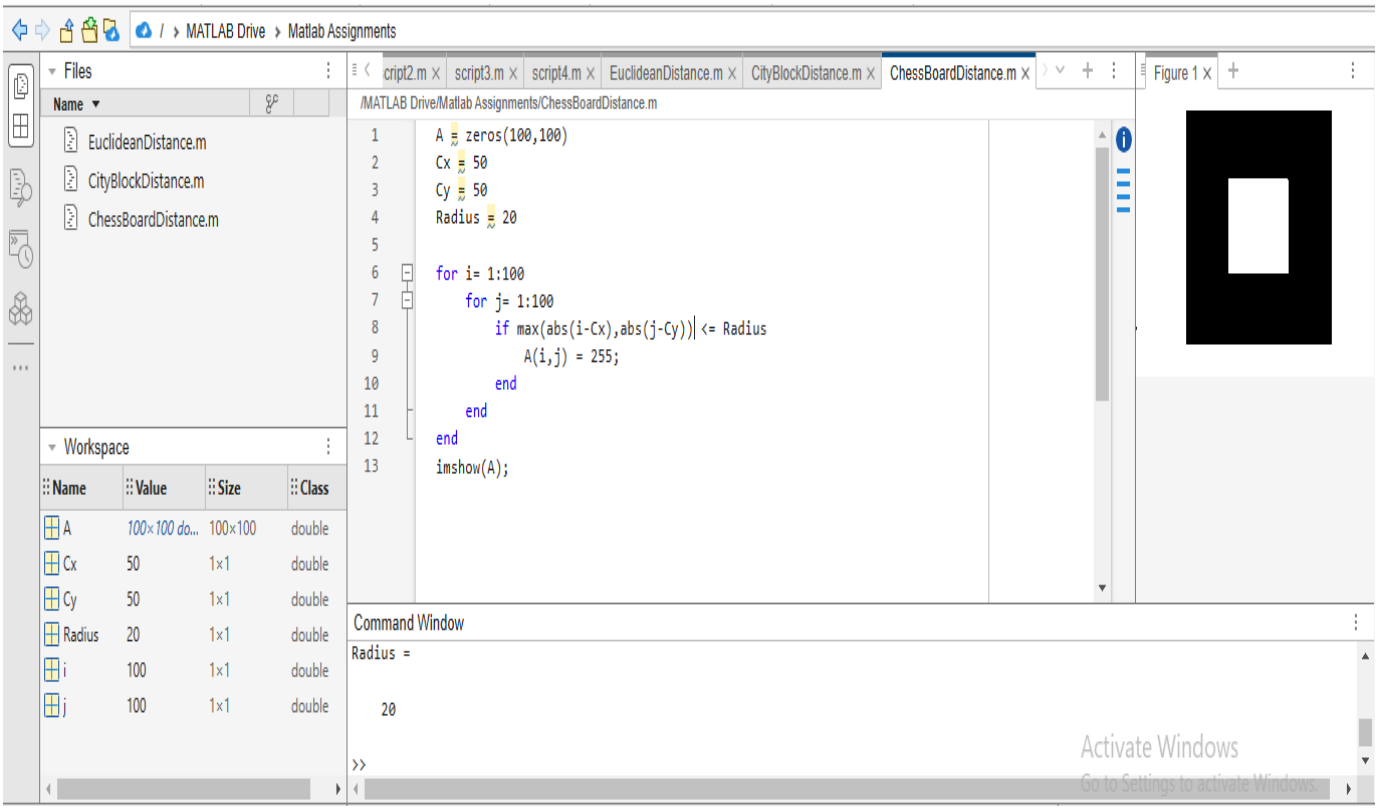


Matlab Code:

```
A = zeros(100, 100);  % Create a 100x100 matrix initialized to zero (black background)
Cx = 50;              % Set the center of the circle on the x-axis
Cy = 50;              % Set the center of the circle on the y-axis
Radius = 20;          % Set the radius of the circle

for i= 1:100
   for j= 1:100
      if abs(i-Cx) + abs(j-Cy)  <= Radius
         A(i,j) = 255;
      end
   end
end
imshow(A);
```

3. Chess Board Distance:



Matlab Code:

```
A = zeros(100,100)
Cx = 50
Cy = 50
Radius = 20

for i= 1:100
   for j= 1:100
      if max(abs(i-Cx),abs(j-Cy)) <= Radius
         A(i,j) = 255;
      end
   end
end
imshow(A);
```

UNIVERSITY OF THE PUNJAB

Gujranwala Campus

DEPARTMENT OF INFORMATION TECHNOLOGY

Assignment #02

Course Title:

" COMPUTER VISION"

Submitted by:

Farheen Akhtar

BIT21029

BS-IT(7th Semester-Morning)

Submitted to:

Mam Fouqia Zafeer

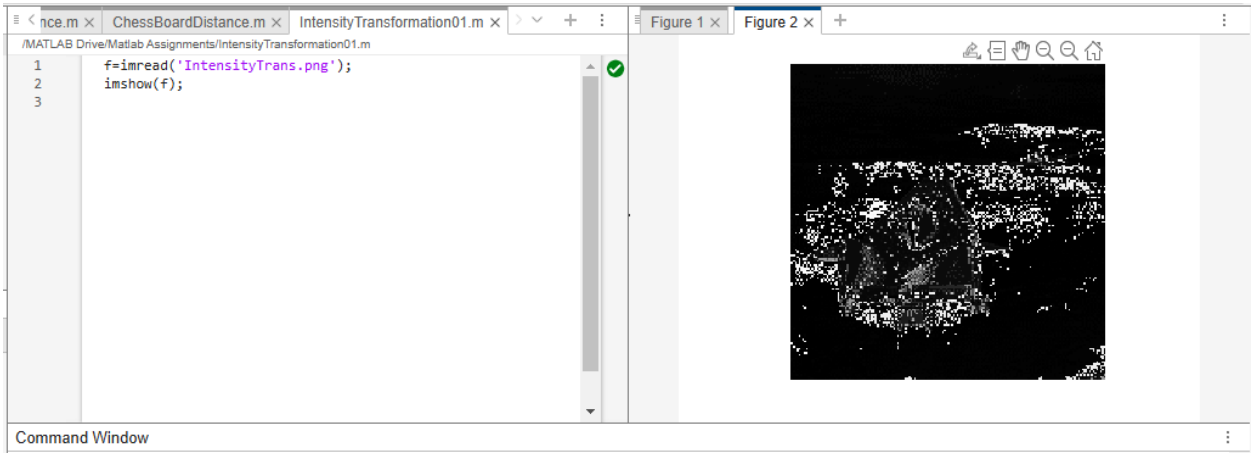
Topic:

“Intensity Transformation”

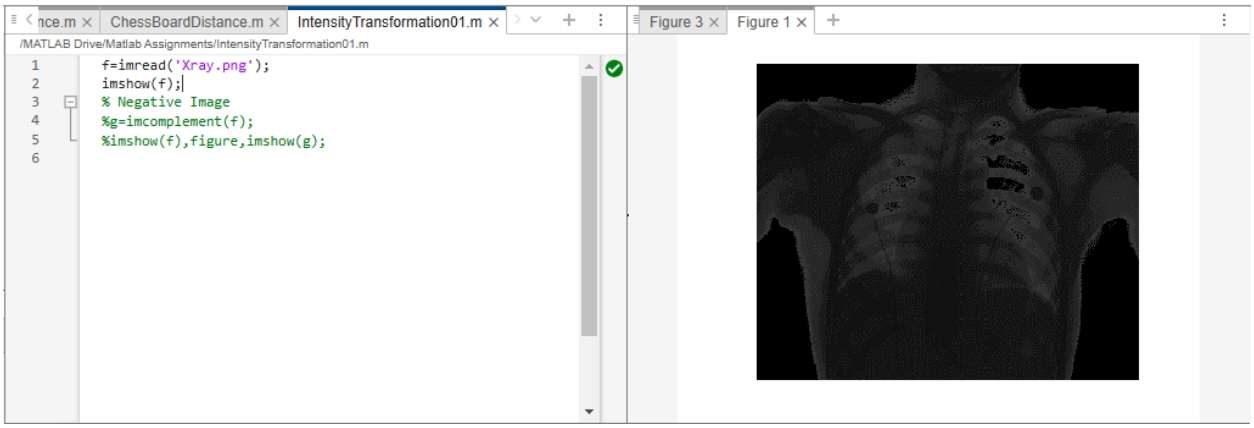
Exercise no: 01

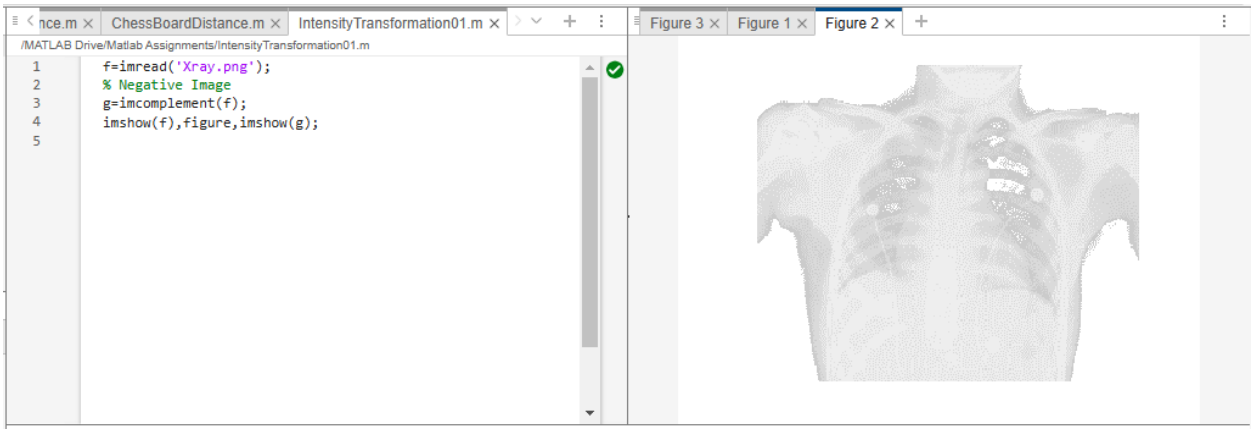
1. Negative Image:

Example#01:



Example#02:





Matlab Code:

```
f=imread('Xray.png');
% Negative Image
g=imcomplement(f);
imshow(f),figure,imshow(g);
```

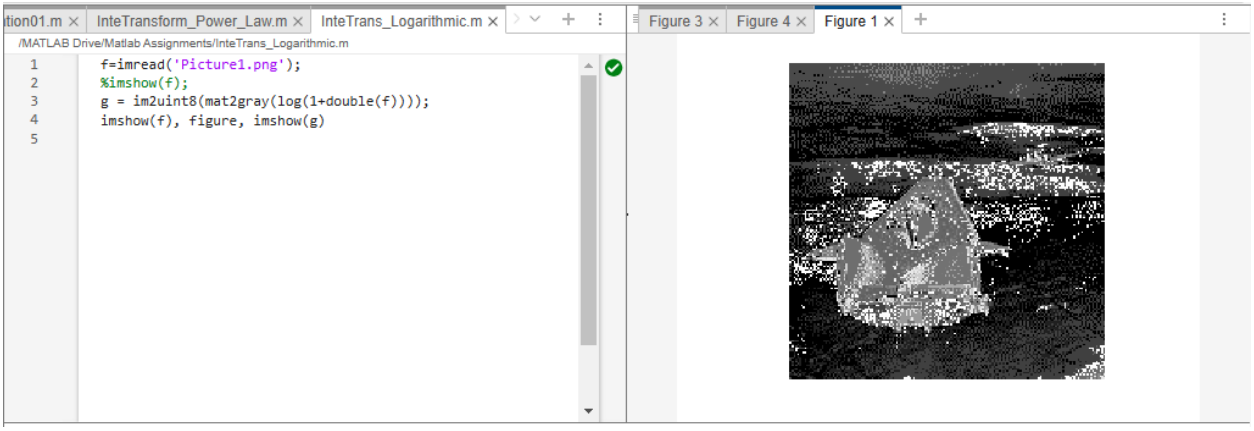
2. Power Law:

Matlab Code:

```
%g = imadjust(f, [low_in high_in], [low_out high_out], gamma);
f=imread('Picture1.png');
%imshow(f);


g = imadjust(f, [ ], [ ], 0.5);
imshow(f), figure, imshow(g)
```

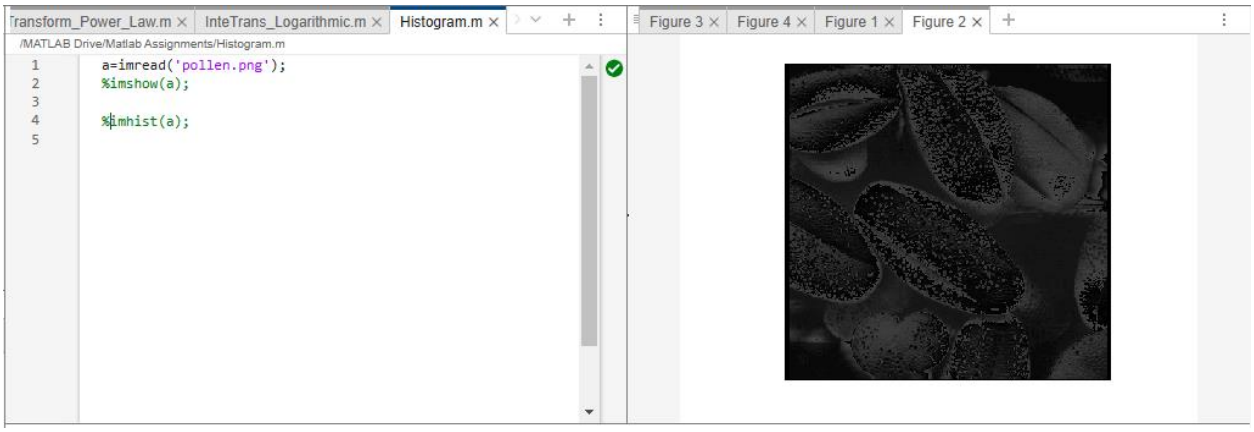
3. Logarithmic:

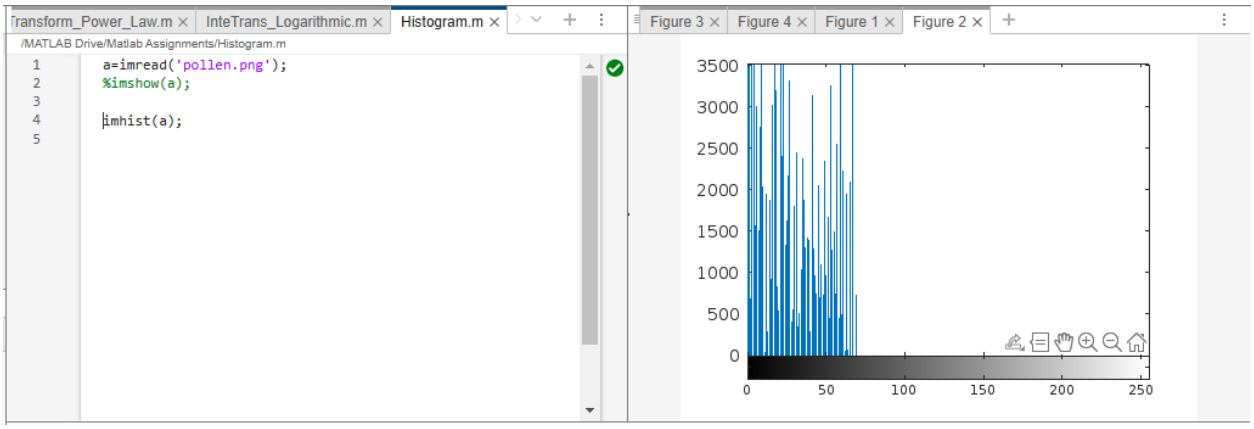


Matlab Code:

```
f=imread('Picture1.png');
%imshow(f);
g = im2uint8(mat2gray(log(1+double(f))));
imshow(f), figure, imshow(g)
```

4. Histogram:





Matlab Code:

```
a=imread('pollen.png');
%imshow(a);

imhist(a);
```

Exercise no: 02

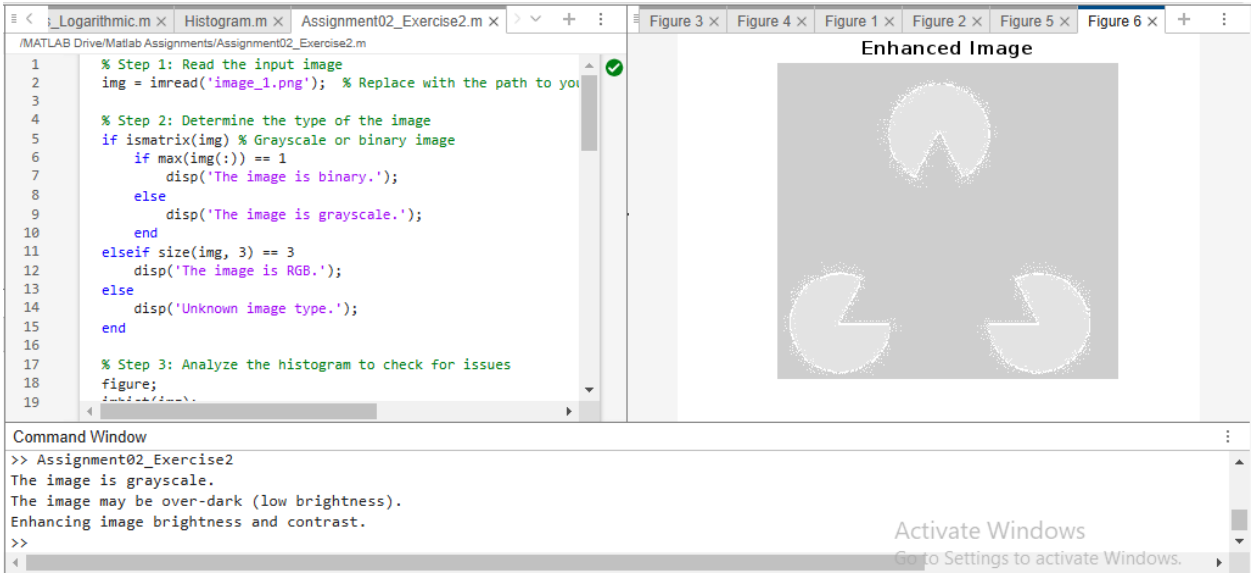
Question:

Write a program which can read an image as an input and do the following automatically. Show the results of all steps.

a) Find the type of image: binary, gray or RGB.

b) Find the issue in image, over dark, over bright, low contrast, or normal. (Hint: can use histogram).

c) Resolve the issue if any and show the final image after enhancement.



ns_Logarithmic.m × | Histogram.m × | Assignment02_Exercise2.m ×

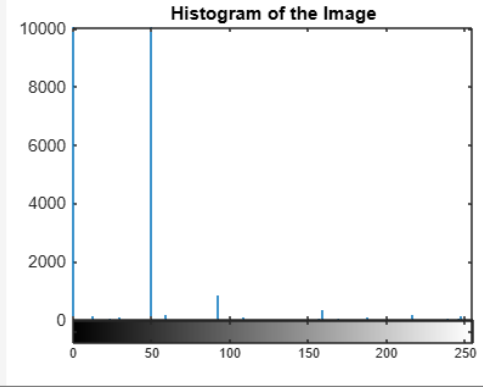
/MATLAB Drive/Matlab Assignments/Assignment02_Exercise2.m

```
% Step 1: Read the input image
img = imread('image_1.png');  % Replace with the path to you

% Step 2: Determine the type of the image
if ismatrix(img) % Grayscale or binary image
    if max(img(:)) == 1
        disp('The image is binary.');
    else
        disp('The image is grayscale.');
    end
elseif size(img, 3) == 3
    disp('The image is RGB.');
else
    disp('Unknown image type.');
end

% Step 3: Analyze the histogram to check for issues
figure;
```

Figure 3 × | Figure 4 × | Figure 1 × | Figure 2 × | Figure 5 × | Figure 6 ×



UNIVERSITY OF THE PUNJAB

Gujranwala Campus

DEPARTMENT OF INFORMATION TECHNOLOGY

Assignment #03

Course Title:

" COMPUTER VISION"

Submitted by:

Farheen Akhtar

BIT21029

BS-IT(7th Semester-Morning)

Submitted to:

Mam Fouqia Zafeer

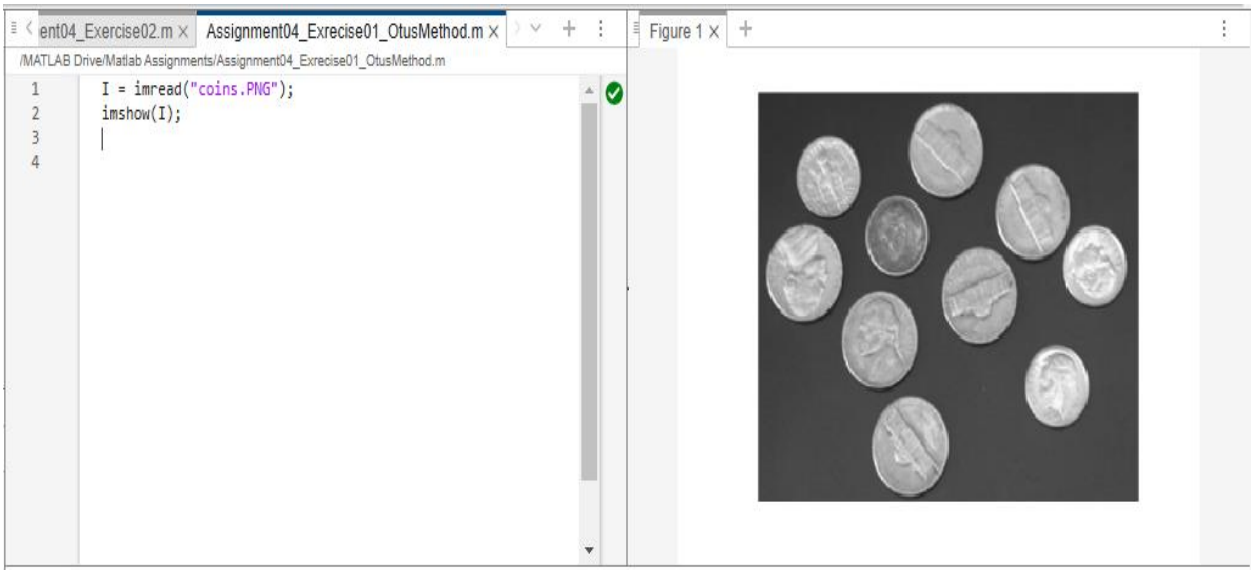
Topic:

“Image Segmentation”

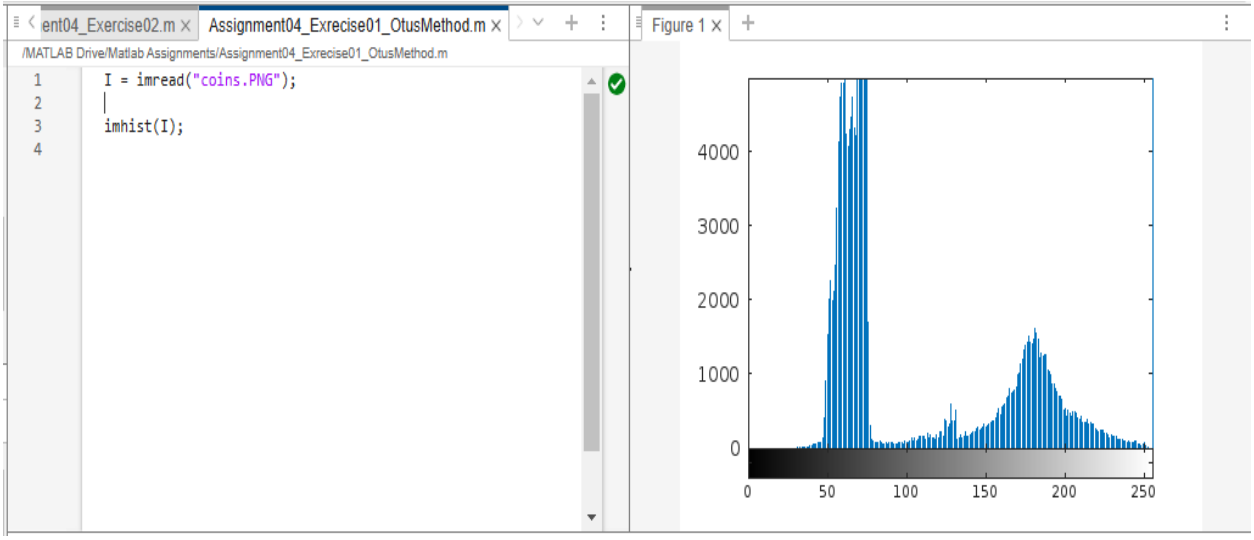
Question no: 1

“Otsu Method”

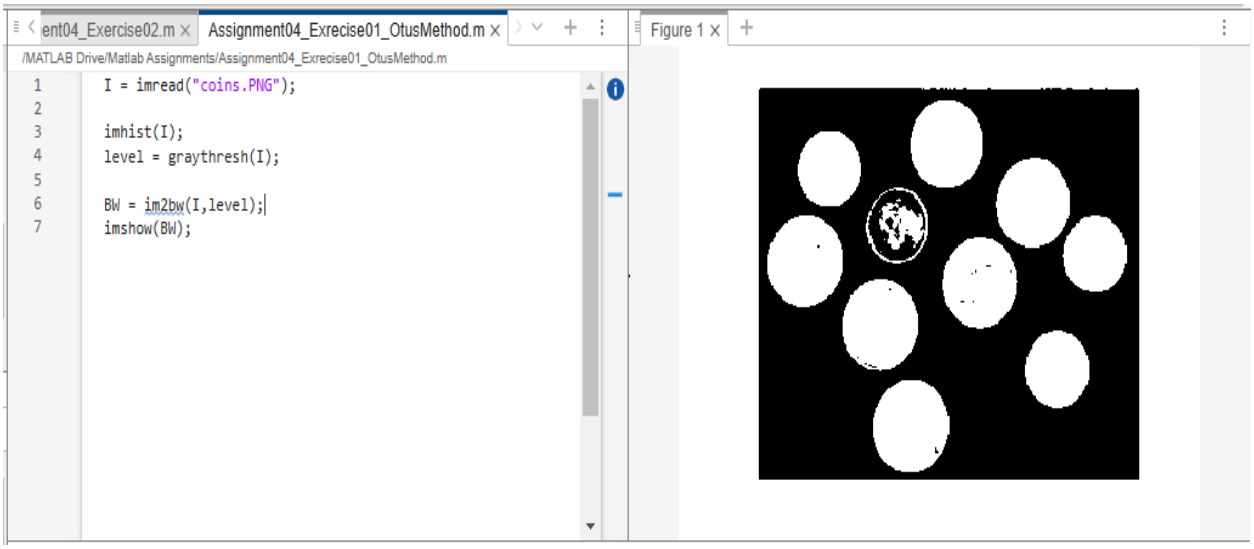
Original image:



Histogram:



Output Image:



Matlab Code:

```
I = imread("coins.PNG");
imhist(I);
level = graythresh(I);
BW = im2bw(I,level);
imshow(BW);
```

UNIVERSITY OF THE PUNJAB

Gujranwala Campus

DEPARTMENT OF INFORMATION TECHNOLOGY

Assignment #04

Course Title:

" COMPUTER VISION"

Submitted by:

Farheen Akhtar

BIT21029

BS-IT(7th Semester-Morning)

Submitted to:

Mam Fouqia Zafeer

Topic:

“Feature Detection”

Question: Detect the largest round object.

Matlab Code:

```
I = imread('Ass3Image1.PNG');

if size(I, 3) == 3
    I = rgb2gray(I);
end

[centers, radii, metric] = imfindcircles(I, [20 100], 'Sensitivity', 0.92);

if ~isempty(radii)
    [~, idx] = max(radii);
    largest_center = centers(idx, :);
    largest_radius = radii(idx);

    imshow(I);
    hold on;
    viscircles(largest_center, largest_radius, 'EdgeColor', 'r', 'LineWidth', 2);
    hold off;
else
    disp('No circles detected.');
end
```

Question 02:

Create a picture of at least two different fruits or vegetables available at your home. Use this picture and names of items in this picture as the input of code which can do the following:

a. Identify the items in picture and label accordingly.

b. Test your code on more than two types of fruits/vegetables for the bonus marks.

Matlab Code:

```
% Read the image
I = imread('Banana_Orange.PNG');

% Convert image to HSV color space
I_hsv = rgb2hsv(I);

% Define color thresholds for banana (yellow) and orange
banana_mask = (I_hsv(:,:,1) >= 0.10) & (I_hsv(:,:,1) <= 0.20); % Adjusted for yellow
orange_mask = (I_hsv(:,:,1) >= 0.01) & (I_hsv(:,:,1) <= 0.10); % Adjusted for orange

% Remove small noise
banana_mask = bwareaopen(banana_mask, 1500);
orange_mask = bwareaopen(orange_mask, 1500);

% Label connected components
[banana_labels, num_bananas] = bwlabel(banana_mask);
[orange_labels, num_oranges] = bwlabel(orange_mask);

% Get region properties
banana_stats = regionprops(banana_labels, 'Centroid', 'Area');
orange_stats = regionprops(orange_labels, 'Centroid', 'Area');
```

```
% Display the original image
imshow(I);
hold on;

% Find and label the largest banana
if num_bananas > 0
    [~, max_idx] = max([banana_stats.Area]); % Find largest banana
    centroid = banana_stats(max_idx).Centroid;
    text(centroid(1), centroid(2), 'Banana', 'Color', 'y', 'FontSize', 14, 'FontWeight', 'bold',
'HorizontalAlignment', 'center');
end

% Find and label the largest orange
if num_oranges > 0
    [~, max_idx] = max([orange_stats.Area]); % Find largest orange
    centroid = orange_stats(max_idx).Centroid;
    text(centroid(1), centroid(2), 'Orange', 'Color', 'r', 'FontSize', 14, 'FontWeight', 'bold',
'HorizontalAlignment', 'center');
end

hold off;
```