INDEX

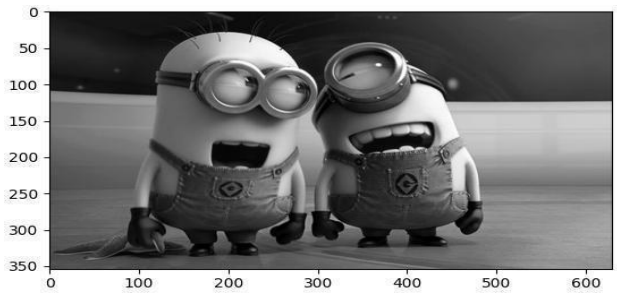
PRACT NO.	PRACTICAL	SIGN
1.	A. Display Of An Image And Conversion Of Gray Scale B. Negative Transform Of An Image	
2.	A. To Perform Image Negation B. Log Trans C. Bit Plane Slicing D. Histogram	
3.	A. High Pass B. Low Pass C. Discrete Fourier Transform	
4.	A. Mean Filter B. Median Filter C. Histogram	
5.	A. Gray To False Color B. Gray Level With Bg C. Gray To Color	
6.	To implement Morphological operations	
7.	To implement Boundary	
8.	To perform Addition and subtraction on two image.	

PRACTICAL 1

A. DISPLAY OF AN IMAGE AND CONVERSION OF GRAY SCALE

```
import PIL
import matplotlib.pyplot as plt
img = PIL.Image.open("image.jpg")
gray_img = img.convert("L")
plt.imshow(gray_img, cmap='gray')
plt.savefig("gray_pic.jpg")
```

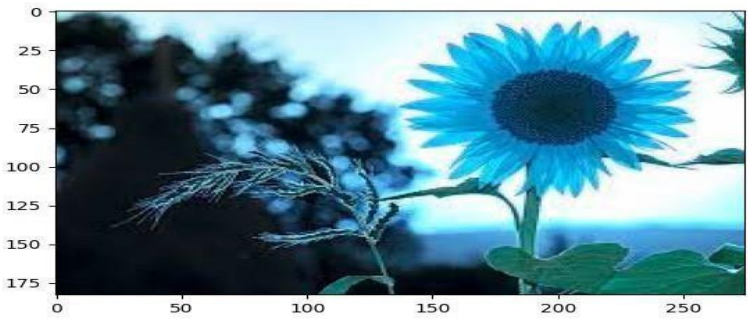
OUTPUT:

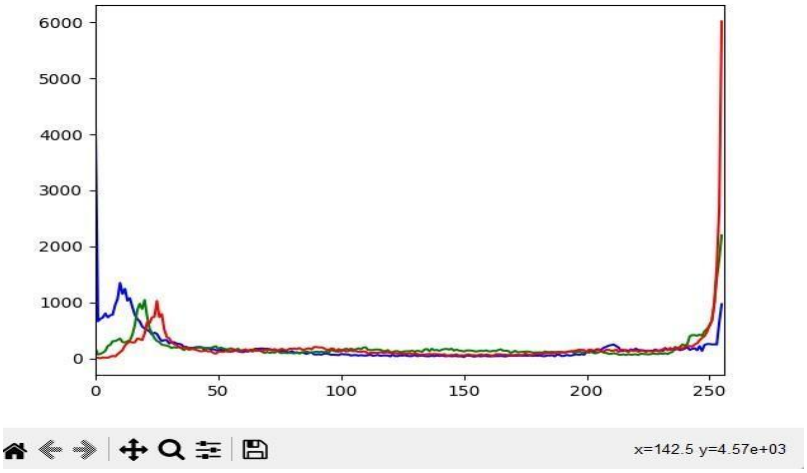


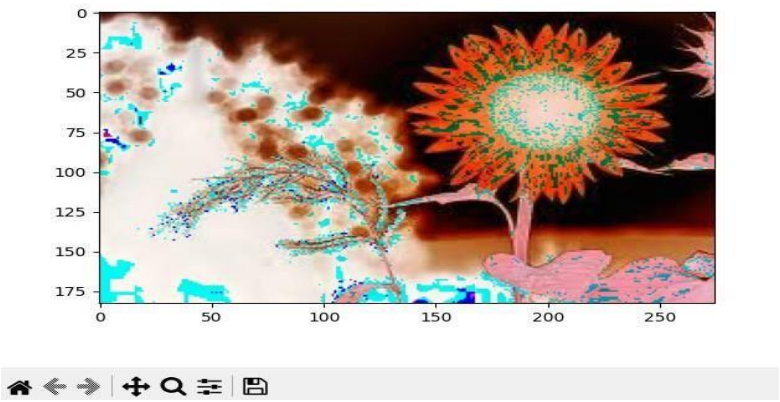
B. NEGATIVE TRANSFORM OF AN IMAGE

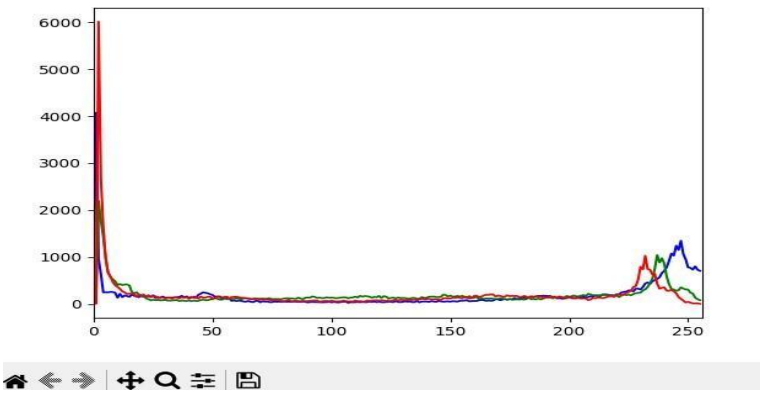
```
import cv2
import matplotlib.pyplot as plt
img_bgr = cv2.imread('image.jpg', 1)
plt.imshow(img_bgr)
plt.show()
color = ('b', 'g', 'r')
for i, col in enumerate(color):
    histr = cv2.calcHist([img_bgr],[i], None,[256],[0, 256])
    plt.plot(histr, color = col)
    plt.xlim([0, 256])
plt.show()
img_neg = 1 - img_bgr
plt.imshow(img_neg)
plt.show()
color = ('b', 'g', 'r')
for i, col in enumerate(color):
    histr = cv2.calcHist([img_neg],[i], None, [256],[0, 256])
    plt.plot(histr, color = col)
    plt.xlim([0, 256])
plt.show()
```

OUTPUT:


0
25
50
75
100
125
150
175
0
50
100
150
200
250


6000
5000
4000
3000
2000
1000
0
0
50
100
150
200
250
x=142.5 y=4.57e+03


0
25
50
75
100
125
150
175
0
50
100
150
200
250

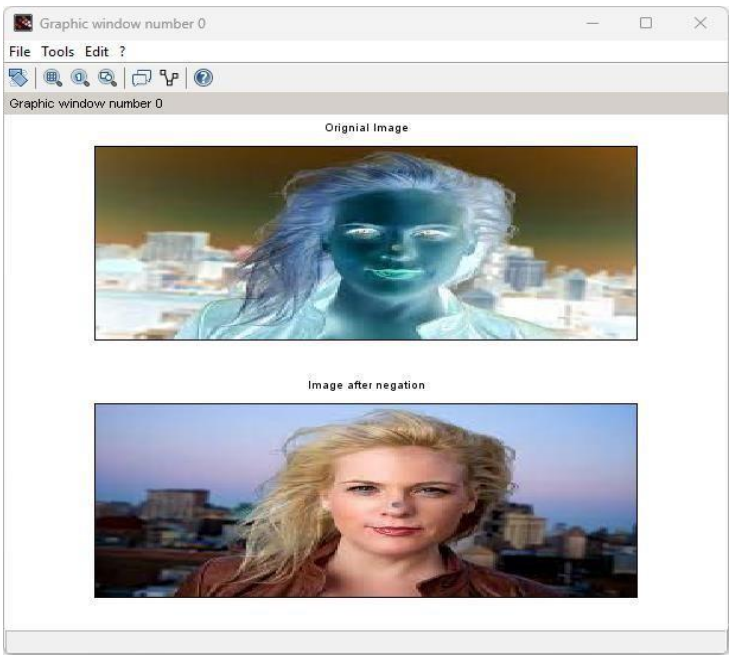

6000
5000
4000
3000
2000
1000
0
0
50
100
150
200
250

PRACTICAL 2

A) TO PERFORM IMAGE NEGATION

```
clc;
clear all;
A
=imread("C:\Users\Student\Desktop\NA\negimg.jpg");
subplot(2,1,1);
imshow(A);
title('Orignial Image');
R = A(:,:,1);
G = A(:,:,2);
B = A(:,:,3);
[row col]=size(A);
for x=1:row
   for y=1:col
      R(x,y)=255-R(x,y);
      G(x,y)=255-G(x,y);
      B(x,y)=255-B(x,y);
   end
end
A(:,:,1)=R;
A(:,:,2)=G;
A(:,:,3)=B;
subplot(2,1,2);
imshow(A);
title('Image after negation');
```

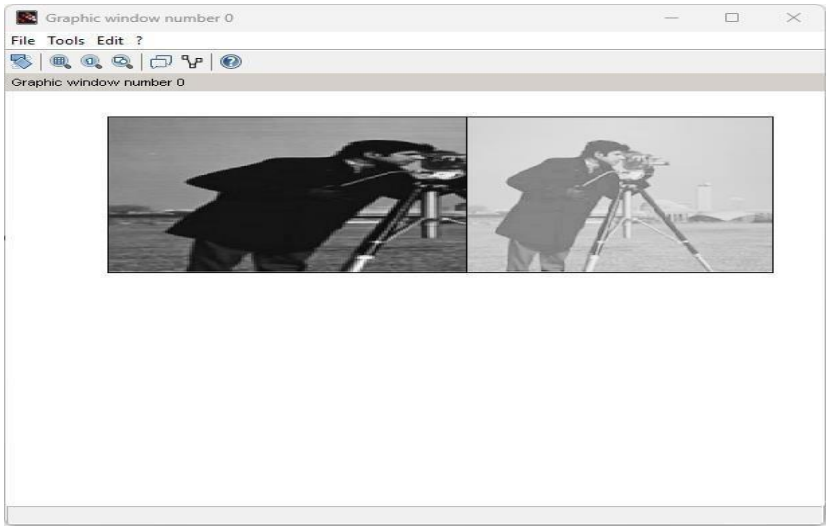
OUTPUT:



B) LOG TRANS

```
clc;
clear all;
a=imread("C:\Users\Student\Downloads\camera.png");
a=rgb2gray(a);
subplot(2,1,1);
imshow(a);
c=1;
[r1,c1]=size(a);
for i=1:r1
   for j=1:c1
      b=double(a(i,j));
      s(i,j)=c*log10(1+b);
   end
end
new1=uint8(s*100);
//imshow(new1);
subplot(2,2,2);
imshow(new1);
```

OUTPUT:

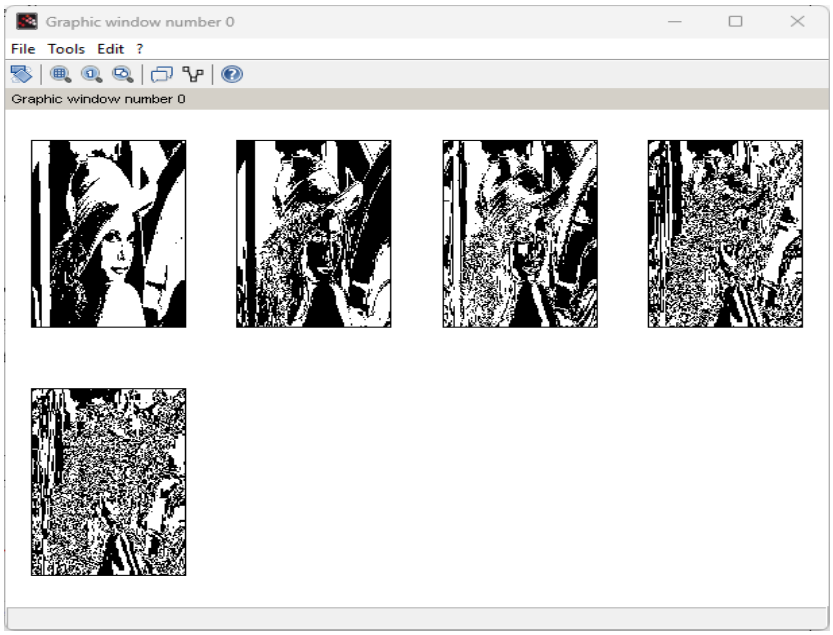


C) BIT PLANE SLICING

```
clc;
clear all;
f=imread("C:\Users\Student\Downloads\lenag.jpeg");
f=double(f);
[r,c]=size(f);
com=[128 64 32 16 8 4 2 1];
for k=1:1:length(com);
   for i=1:r
      for j=1:c
      new(i,j)=bitand(f(i,j),com(k));
   end
```

```
    subplot(2,4,k);
    imshow(new);
    end
end
```

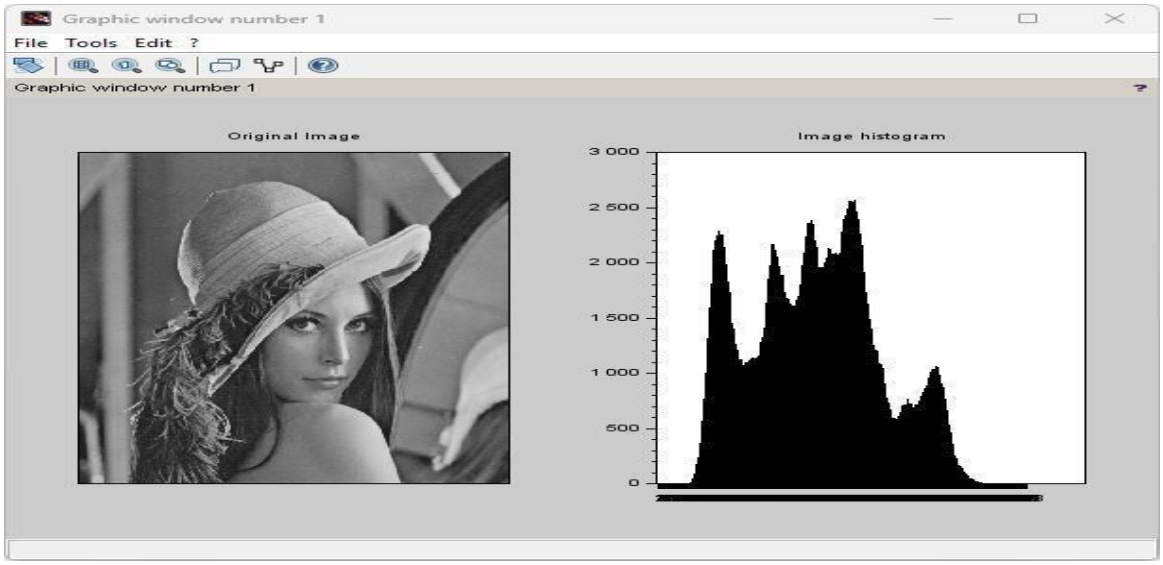
OUTPUT:



D) HISTOGRAM

```
clc;
clear all;
a=imread('C:\Users\Student\Downloads\lena.jpeg');
a=rgb2gray(a);
h=zeros(1,258);
[r,c]=size(a);
for i=1:r
    for j=1:c
        if (a(i,j)==0)
            h(0)=h(0)+1;
        end
        k=a(i,j);
        h(k)=h(k)+1;
    end
end
figure(1);
subplot(1,2,1);
imshow(uint8(a));
title('Original Image')
subplot(1,2,2);
bar(h);
title('Image histogram');
```

OUTPUT:

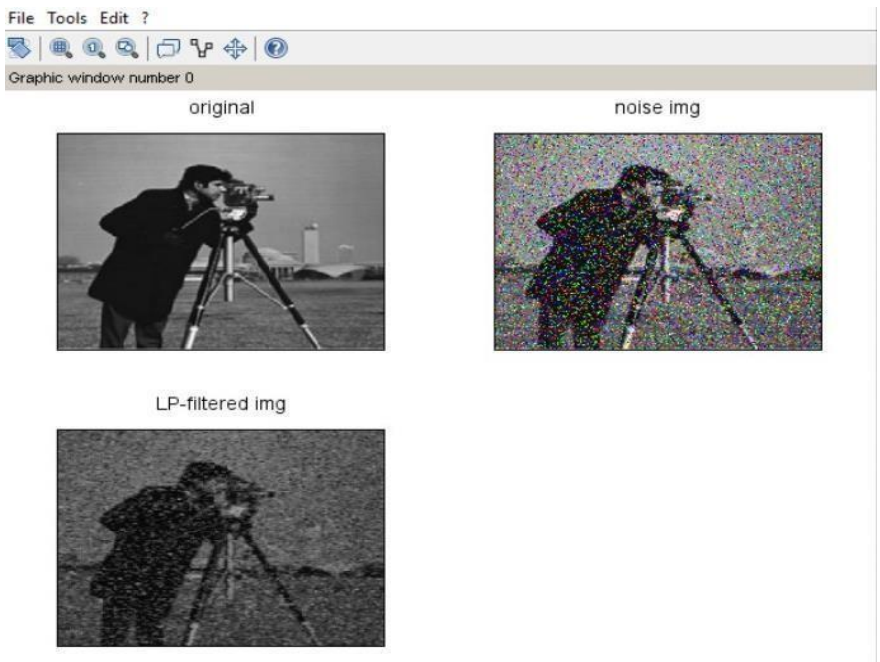

Graphic window number 1
File Tools Edit ?
Graphic window number 1
Original Image
Image histogram
3 000
2 500
2 000
1 500
1 000
500
0

PRACTICAL 3

A. HIGH PASS

```
a=imread('C:\Users\Student\Downloads\NA\OIP.jpg'
);b=double(a);
c=imnoise(a,'salt & pepper',0.2);
d=double(c);
subplot(2,2,1);
imshow(a);
title('original');
subplot(2,2,2);
imshow(c);
title('noise img');
m=[-1 -1 -1;-1 8 -1;-1 -1 -1];
[r1,c1]=size(a);
for i=2:1:r1-1
    for j=2:1:c1-1
   new(i,j)=(m(1)*d(i-1,j-1))+(m(2)*d(i-1,j))+(m(3)*d(i-1,j+1))    +(m(4)*d(i,j
1))+(m(5)*d(i,j))+(m(6)*d(i,j+1)) +(m(7)*d(i+1,j-1))+(m(8)*d(i+1,j))+(m(9)*d(i+1,j+1));
    end
end
subplot(2,2,3);
imshow(uint8(new));
title('LP-filtered img');
```

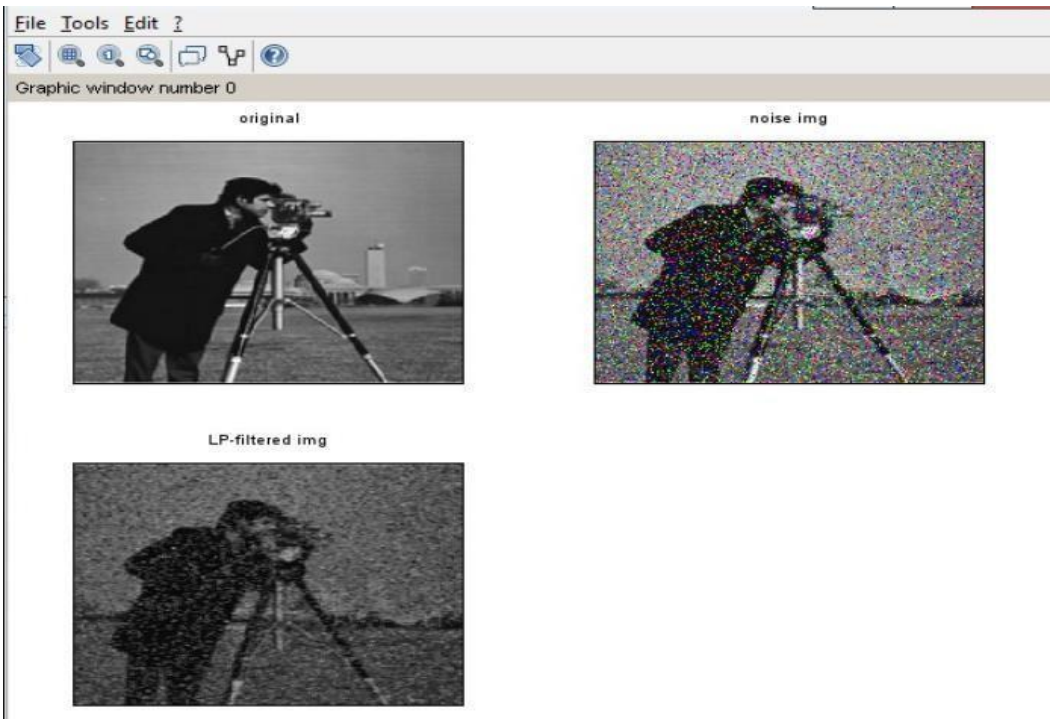
OUTPUT:



B. LOW PASS

```
clc;
clear all;
a=imread('C:\Users\Student\Downloads\NA\camera (1)
(1).png');b=double(a);
c=imnoise(a,'salt & pepper',0.2);
d=double(c);
subplot(2,2,1);
imshow(a);
title('original');
subplot(2,2,2);
imshow(c);
title('noise img');
m=(1/9)*(ones(3,3));
[r1,c1]=size(a);
for i=2:1:r1-1
    for j=2:1:c1-1
  new(i,j)=(m(1)*d(i-1,j-1))+(m(2)*d(i-1,j))+(m(3)*d(i-1,j+1))    +(m(4)*d(i,j
1))+(m(5)*d(i,j))+(m(6)*d(i,j+1))  +(m(7)*d(i+1,j-
1))+(m(8)*d(i+1,j))+(m(9)*d(i+1,j+1));
    end
end
subplot(2,2,3);
imshow(uint8(new));
title('LP-filtered img');
```

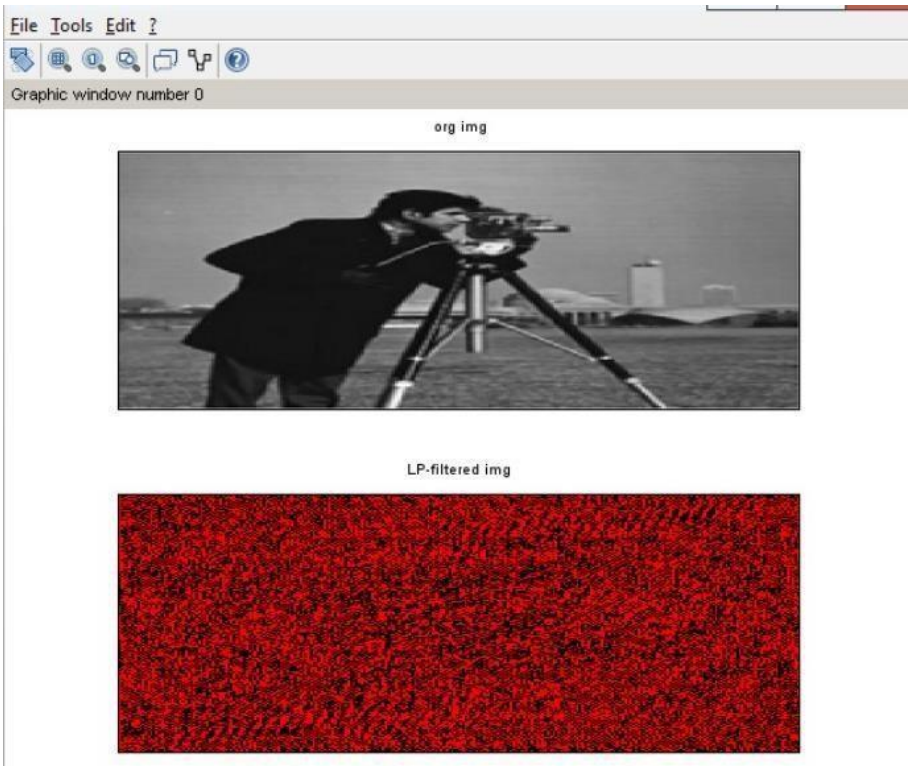
OUTPUT:



C. DISCRETE FOURIER TRANSFORM

```
a=imread('C:\Users\Student\Downloads\camera (1).png');
subplot(2,1,1);
imshow(a);
title('org img');
b=double(a);
c=fft(b);
subplot(2,1,2);
imshow(c);
title('LP-filtered img');
```

OUTPUT:

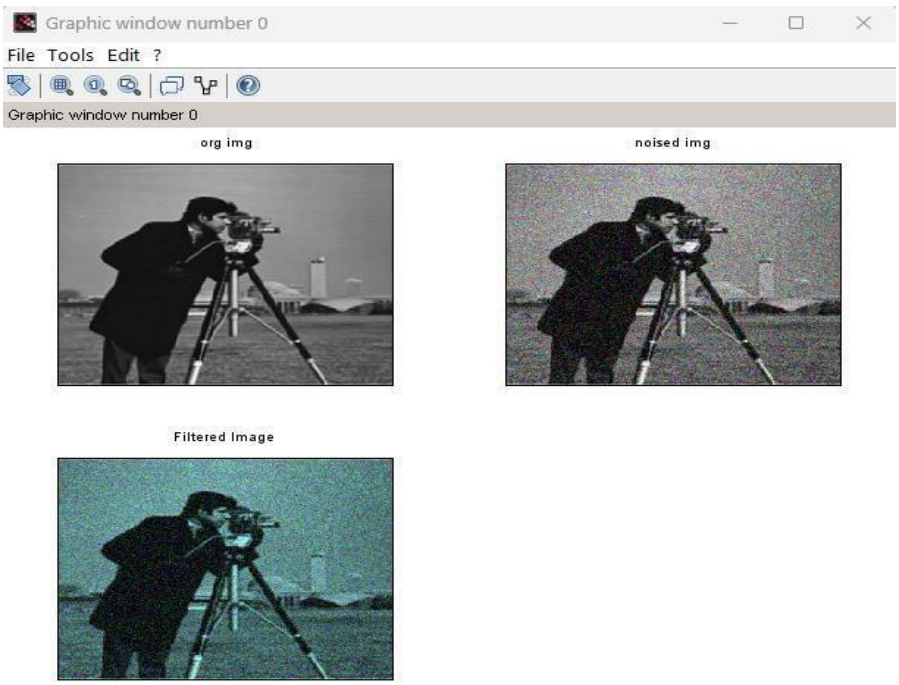


PRACTICAL 4

A) MEAN FILTER

```
a=imread('C:\Users\Student\Downloads\NA\camera (1)
(1).png');b1=double(a);
c=imnoise(a,'gaussian');
d=double(c);
b=d;
m=(1/9)*(ones(3,3));
[r1,c1]=size(a);
subplot(2,2,1);
imshow(a);
title('org img');
subplot(2,2,2);
imshow(c);
title('noised img');
for i=2:r1-1
for j=2:c1-1
a1=d(i-1,j-1)+d(i-1,j)+d(i-1,j+1)+d(i,j-1)+d(i,j)+d(i,j+1)+d(i+1,j-1)+d(i+1,j)+d(i+1,j+1);
b(i,j)=a1*(1/9);
end
end
subplot(2,2,3);
imshow(uint8(b));
title('Filtered Image');
clc;
clear all;
```

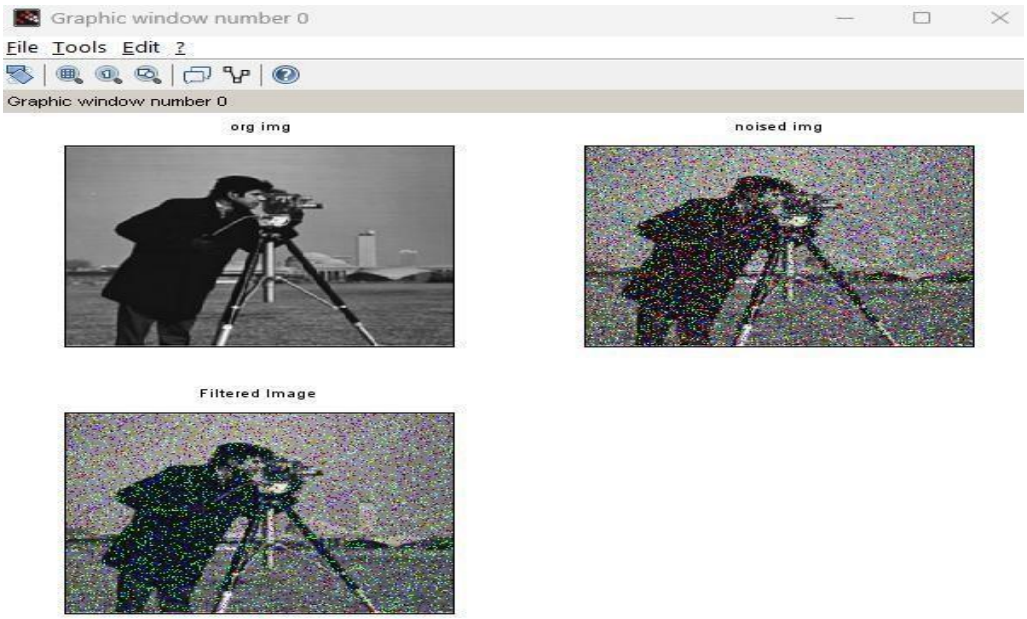
OUTPUT:



B) MEDIAN FILTER

```
a=imread("C:\Users\Student\Downloads\NA\camera (1)
(1).png');b1 = double(mtlb_double(a));
c = imnoise(a,"salt & pepper",0.2);
d = double(mtlb_double(c));
b = d;
m = (1/9)*ones(3,3);
subplot(2,2,1);
imshow(a);
title('org img');
subplot(2,2,2);
imshow(c);
title('noised img');
[r1,c1] = size(mtlb_double(a));
for i = 2:r1-1
      for j = 2:c1-1
      a1 = [d(i-1,j-1),d(i-1,j),d(i-1,j+1),d(i,j-1),d(i,j), d(i,j+1),d(i+1,j1),d(i+1,j),d(i+1,j+1)];
      a2 = gsort(a1,"g","i");//gsort(A,'g','i') sort the elements of the array A in the
increasing order.
      med = a2(5);
      b(i,j) = med;
      end;
end;
subplot(2,2,3);
imshow(uint8(b));
title('Filtered Image');
```

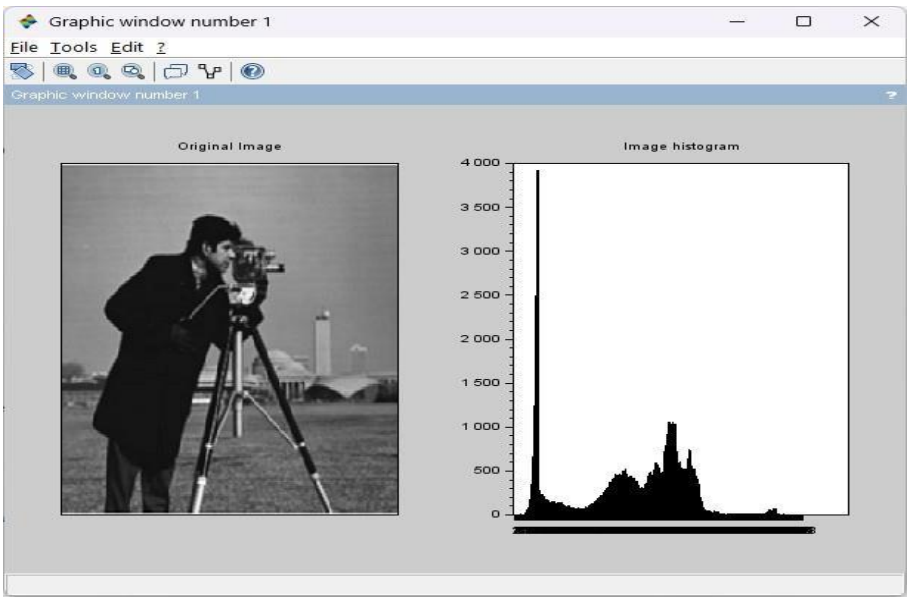
OUTPUT:



C) HISTOGRAM

```
clc;
clear all;
a=imread('C:\Users\Student\Downloads\NA\camera (1)
(1).png');a=rgb2gray(a);
h=zeros(1,258);
[r,c]=size(a);
for i=1:r
   for j=1:c
      if (a(i,j)==0)
         h(0)=h(0)+1;
      end
      k=a(i,j);
      h(k)=h(k)+1;
   end
end
figure(1);
subplot(1,2,1);
imshow(uint8(a));
title('Original Image')
subplot(1,2,2);
bar(h);
title('Image histogram');
```

OUTPUT:



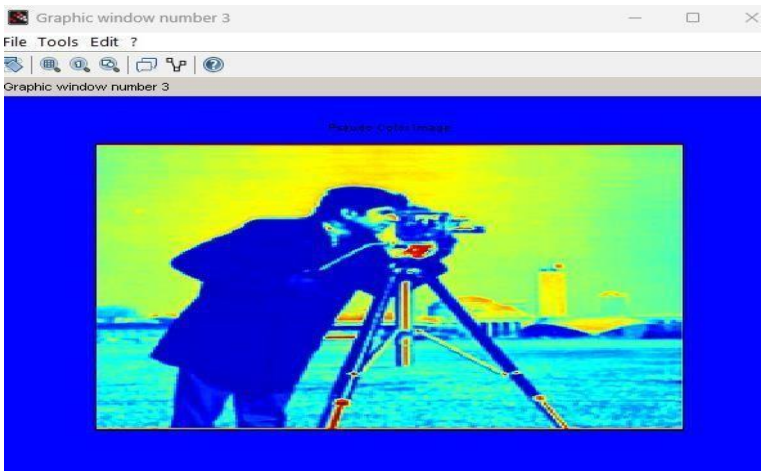
PRACTICAL 5

A) GRAY TO FALSE COLOR

```
clc;


close;
a = imread('C:\Users\Student\Downloads\NA\camera (1) (1).png');
//Displaying Original RGB image
figure(1);
imshow(a);
title("Original Image")
//Displaying Gray level image
b = rgb2gray(a);
figure(2);
imshow(b);
title("Gray Level Image")
//Displaying False coloring(Pseudo) image
figure(3)
imshow(b,jetcolormap(256));
title("Pseudo Color Image");
```

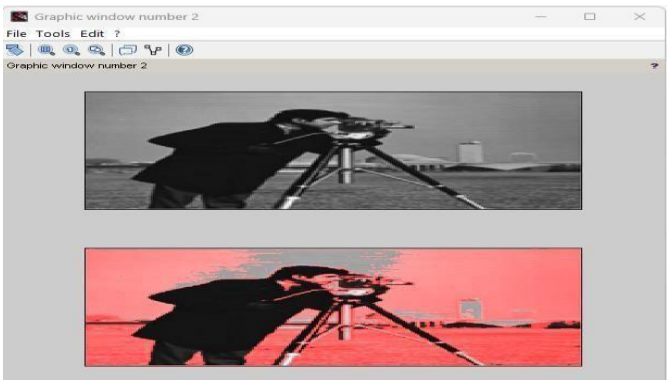
OUTPUT:



B) GRAY LEVEL WITH BG

```
clc;
clear all;
a=imread('C:\Users\Student\Downloads\NA\camera (1)
(1).png');a1=58; // This value is user defined
b1=158; // This value is user defined
[r,c]=size(a);
figure(2);
subplot(2,1,1);
imshow(a);
for i=1:r
   for j=1:c
      if (a(i,j)>a1 & a(i,j)<b1)
         x(i,j)=255;
 else
         x(i,j)=a(i,j);
      end
   end
end
x=uint8(x);
subplot(2,1,2);
imshow(x);
```

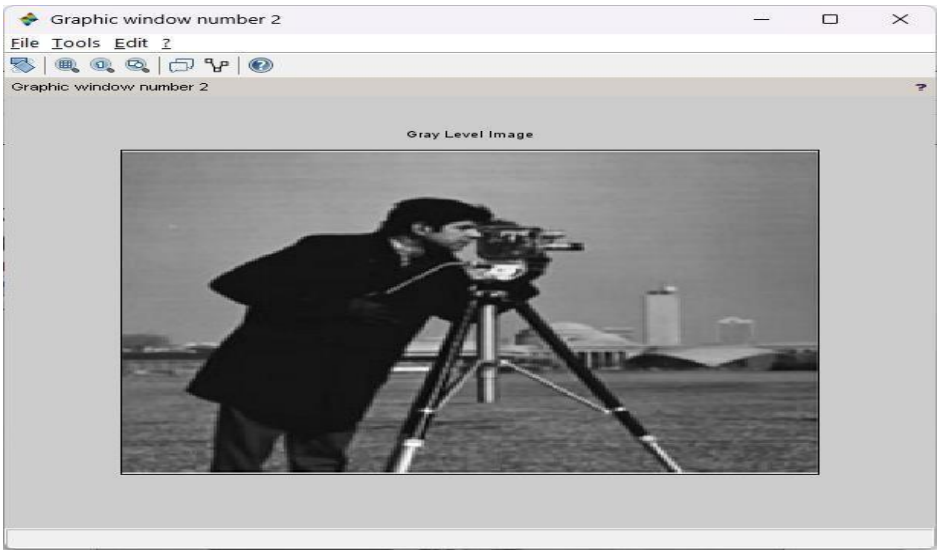
OUTPUT:

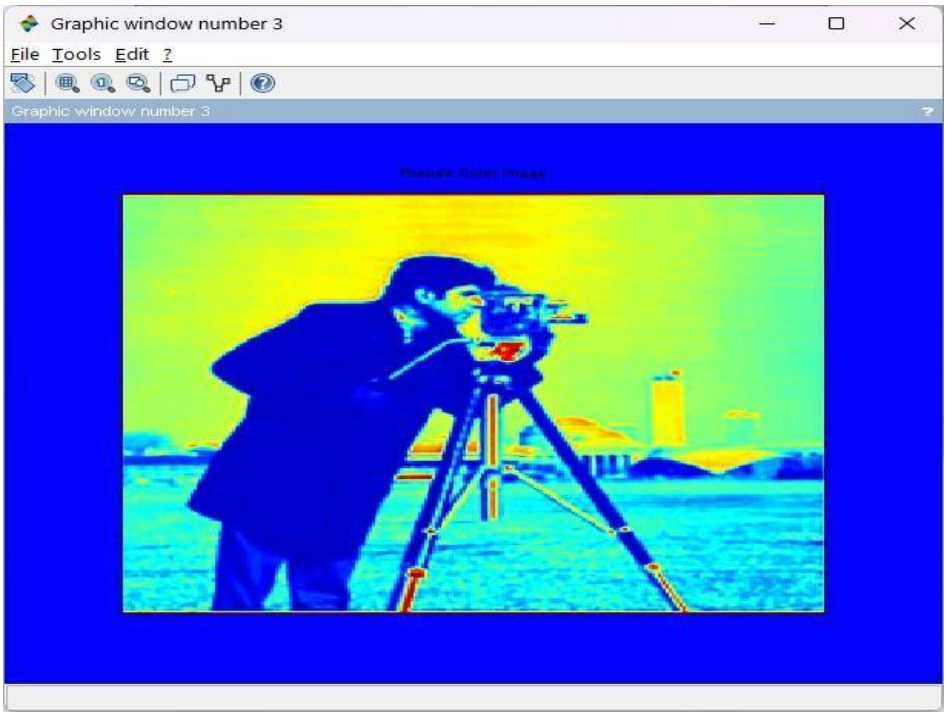


C) GRAY TO COLOR

```
clc;
close;
a = imread('C:\Users\Student\Downloads\NA\camera (1) (1).png');
//Displaying Original RGB image
figure(1);
imshow(a);
title("Original Image")
//Displaying Gray level image
b = rgb2gray(a);
figure(2);
imshow(b);
title("Gray Level Image")
//Displaying False coloring(Pseudo) image
figure(3)
imshow(b,jetcolormap(256));
title("Pseudo Color Image");
```

OUTPUT:


Graphic window number 2
File Tools Edit ?
Graphic window number 2
Gray Level Image


Graphic window number 3
File Tools Edit ?
Graphic window number 3

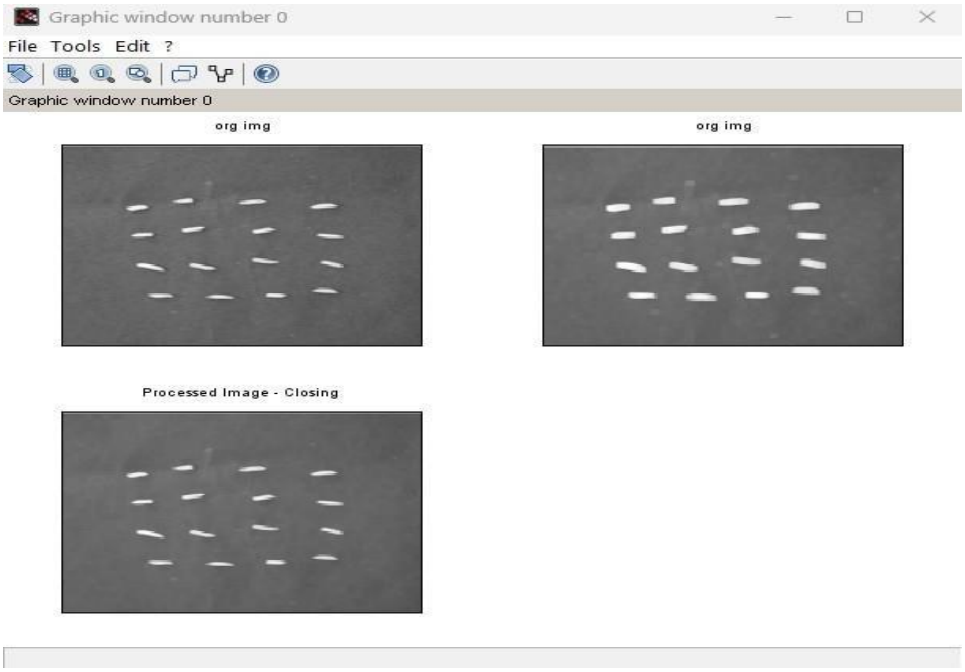
PRATICAL 6

Aim: To implement Morphological operations

A) Closing

```
clc;
clear all;
a=imread('C:\NA\rice.jpeg'
);a=rgb2gray(a);
d=a;
A2=d;
A1=d;
subplot(2,2,1);
imshow(a);
title('org img');
[r,c]=size(d);
m=[1 1 1;1 1 1;1 1 1];
for i=2:1:r-1
for j=2:1:c-1
new=[(m(1)*d(i-1,j-1)) (m(2)*d(i-1,j)) (m(3)*d(i-1,j+1)) (m(4)*d(i,j-1)) (m(5)*d(i,j))
(m(6)*d(i,j+1)) (m(7)*d(i+1,j-1)) (m(8)*d(i+1,j)) (m(9)*d(i+1,j+1))];
A2(i,j)=max(new);
end
subplot(2,2,2);
imshow(A2);
title('org img');
end
d = A2;
A1=A2;
[r,c]=size(d);
for  i=2:1:r-1
for j=2:1:c-1
new=[(m(1)*d(i-1,j-1)) (m(2)*d(i-1,j)) (m(3)*d(i-1,j+1)) (m(4)*d(i,j-1)) (m(5)*d(i,j))
(m(6)*d(i,j+1))(m(7)*d(i+1,j-1)) (m(8)*d(i+1,j)) (m(9)*d(i+1,j+1))];
A1(i,j)=min(new);
end
subplot(2,2,3);
imshow(A1);title('Processed Image - Closing');
end
```

OUTPUT:

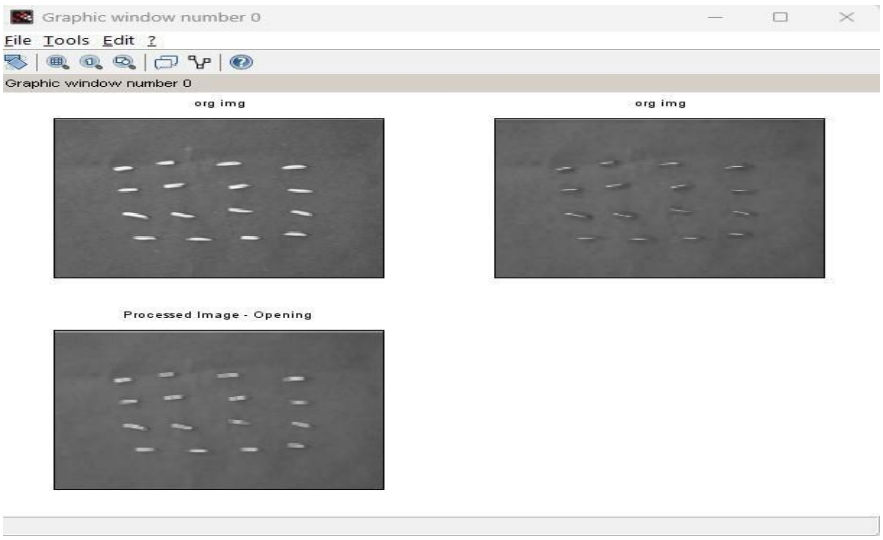


B) Opening

```
clc;
clear all;
a=imread("C:\NA\rice.jpeg"
);a=rgb2gray(a);
d=a;
A2=d;
A1=d;
subplot(2,2,1);
imshow(a);
title('org img');
[r,c]=size(d);
m=[1 1 1;1 1 1;1 1 1];
for i=2:1:r-1
for j=2:1:c-1
new=[(m(1)*d(i-1,j-1)) (m(2)*d(i-1,j)) (m(3)*d(i-1,j+1))
(m(4)*d(i,j-1)) (m(5)*d(i,j)) (m(6)*d(i,j+1))
(m(7)*d(i+1,j-1)) (m(8)*d(i+1,j)) (m(9)*d(i+1,j+1))];
A2(i,j)=min(new);
end
subplot(2,2,2);
imshow(A2);
title('org img');
end
d = A2;
A1=A2;
```

```
[r,c]=size(d);
for  i=2:1:r-1
for j=2:1:c-1
new=[(m(1)*d(i-1,j-1)) (m(2)*d(i-1,j)) (m(3)*d(i-1,j+1))
(m(4)*d(i,j-1)) (m(5)*d(i,j)) (m(6)*d(i,j+1))
(m(7)*d(i+1,j-1)) (m(8)*d(i+1,j)) (m(9)*d(i+1,j+1))];
A1(i,j)=max(new);
end
subplot(2,2,3);
imshow(A1);title('Processed Image - Opening');
end
```

OUTPUT:



C)Dialation

```
clc;
clear all;
a=imread('C:\NA\rectb.png'
);a=rgb2gray(a);
d=a;
A1=a;
[r,c]=size(d);
subplot(2,1,1);
imshow(a);
title('org img');
m=[1 1 1;1 1 1;1 1 1];
// m=ones(5,5);
for i=2:1:r-1
for j=2:1:c-1
```

```
new=[(m(1)*d(i-1,j-1)) (m(2)*d(i-1,j)) (m(3)*d(i-1,j+1)) (m(4)*d(i,j-1)) (m(5)*d(i,j))
(m(6)*d(i,j+1)) (m(7)*d(i+1,j-1)) (m(8)*d(i+1,j)) (m(9)*d(i+1,j+1))];
A1(i,j)=max(new);
end
subplot(2,1,2);
imshow(A1);title('Processed Image - dilation');
end
```

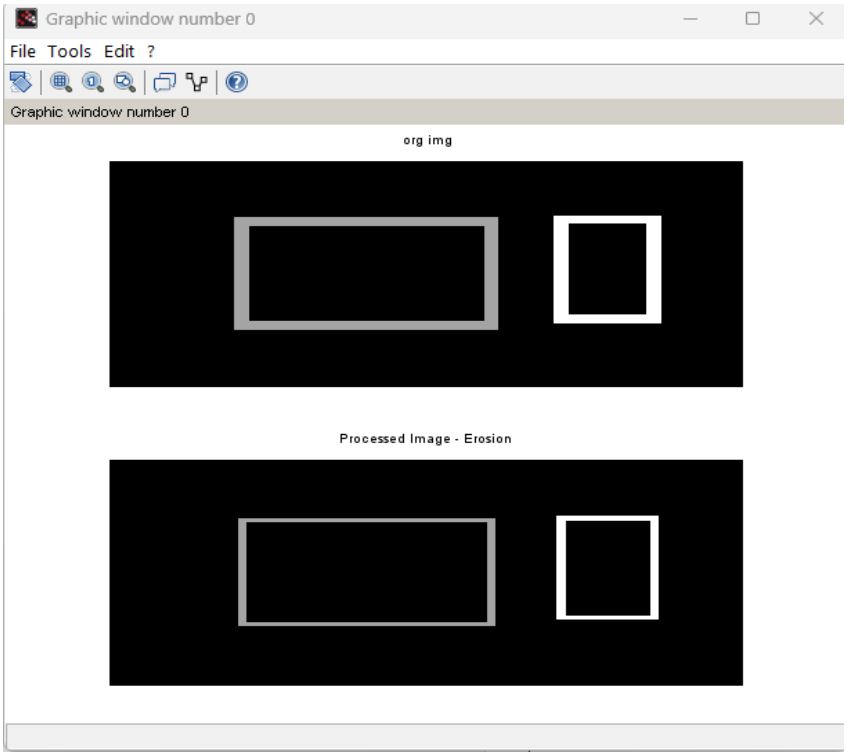
OUTPUT:

D) Erosion

```
clc;
clear all;
a=imread('C:\NA\rectb.png'
);a=rgb2gray(a);
subplot(2,1,1);
imshow(a);
title('org img');
A1=a;
d=a;
[r,c]=size(d);
m=[1 1 1;1 1 1;1 1 1];
// m=ones(5,5);
for i=2:1:r-1
for j=2:1:c-1
   new=[(m(1)*d(i-1,j-1)) (m(2)*d(i-1,j)) (m(3)*d(i-1,j+1)) (m(4)*d(i,j-1))
(m(5)*d(i,j)) (m(6)*d(i,j+1)) (m(7)*d(i+1,j-1)) (m(8)*d(i+1,j)) (m(9)*d(i+1,j+1))];
A1(i,j)=min(new);
end
subplot(2,1,2);
```

```
title('org img');imshow(A1);title('Processed Image - Erosion');
end
```

OUTPUT:

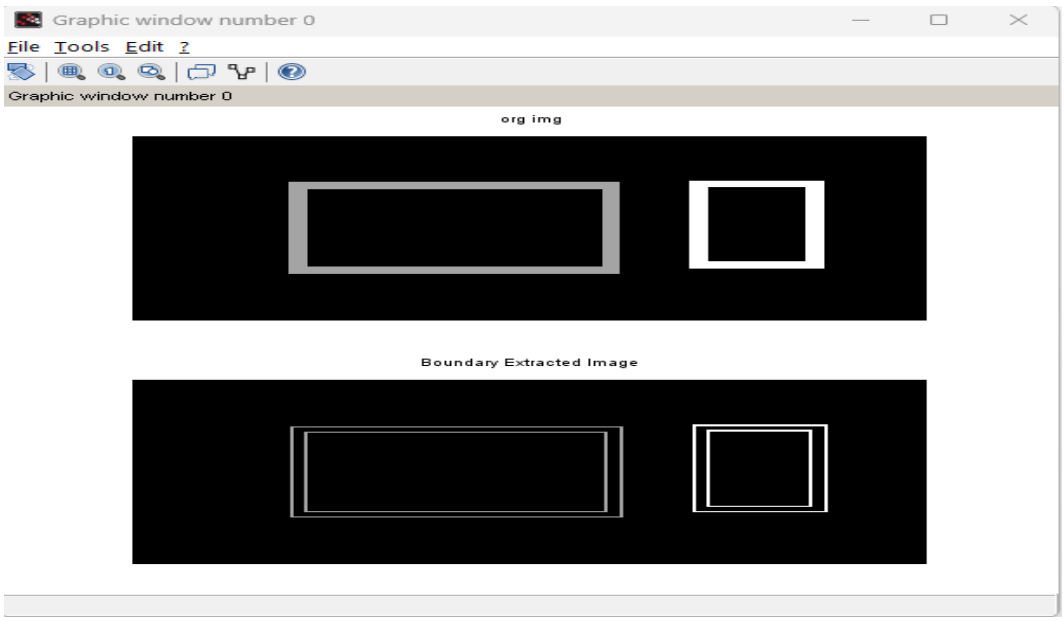


PRACTICAL 7

Aim : To implement Boundary

```
clc;
clear all;
a=imread('C:\NA\rectb.png'
);a=rgb2gray(a);
subplot(2,1,1);
imshow(a);
title('org img');
d=a;
[r,c]=size(d);
m=[1 1 1;1 1 1;1 1 1];
for i=2:1:r-1
for j=2:1:c-1
new=[(m(1)*d(i-1,j-1)) (m(2)*d(i-1,j)) (m(3)*d(i-1,j+1))
(m(4)*d(i,j-1)) (m(5)*d(i,j)) (m(6)*d(i,j+1))
(m(7)*d(i+1,j-1)) (m(8)*d(i+1,j)) (m(9)*d(i+1,j+1))];
A2(i,j)=min(new);
aa(i,j)=d(i,j)-A2(i,j);
end
end
subplot(2,1,2);
imshow(aa);title('Boundary Extracted Image');
```

OUTPUT:



PRATICAL 8

Aim: To perform Addition and subtraction on two image

```
clc;
clear all;
A=imread('C:\NA\circle.png');
B=imread('C:\NA\camera
(2).png');A=rgb2gray(A);
B=rgb2gray(B);
C=imadd(B, A);
D=imsubtract(B, A);
figure(1);
subplot(2,2,1);
imshow(A);
subplot(2,2,2);
imshow(B);
subplot(2,2,3);
imshow(C);
subplot(2,2,4);
imshow(D);
```

OUTPUT: