

DIP PRACTICAL

Practical no: 1

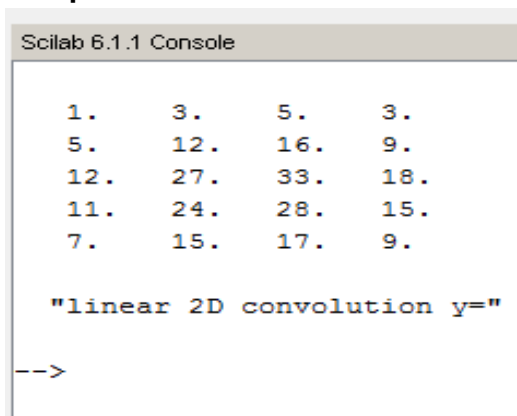
AIM:- 2D Linear Convolution, Circular Convolution between two 2D matrices.

a) : 2D Linear Convolution

sources code:

```
clc;
x=[1,2,3;4,5,6;7,8,9];
h=[1,1;1,1;1,1];
y=conv2(x,h);
disp(y,'linear 2D convolution y=');
```

output:



```
Scilab 6.1.1 Console

1.    3.    5.    3.
5.    12.   16.    9.
12.   27.   33.   18.
11.   24.   28.   15.
7.    15.   17.    9.

"linear 2D convolution y="

-->
```

b) Circular Convolution

sources code:

```
clc ;
x = input("Enter the values of x(n)");
h = input("Enter the values of h(n)");
X = fft2(x);
```

```

H = fft2(h);
Y = X.*H;
y = ifft(Y);
disp (y, 'Circular Convolution Result y = ');

```

output:

```

Scilab 6.1.1 Console

Enter the values of x(n) [1,2;3,4]

Enter the values of h(n) [5,6;7,8]

70.    68.
62.    60.

"Circular Convolution Result y = "

-->

```

PRACTICAL NO : 2

AIM :- Circular Convolution expressed as linear convolution plus alias.

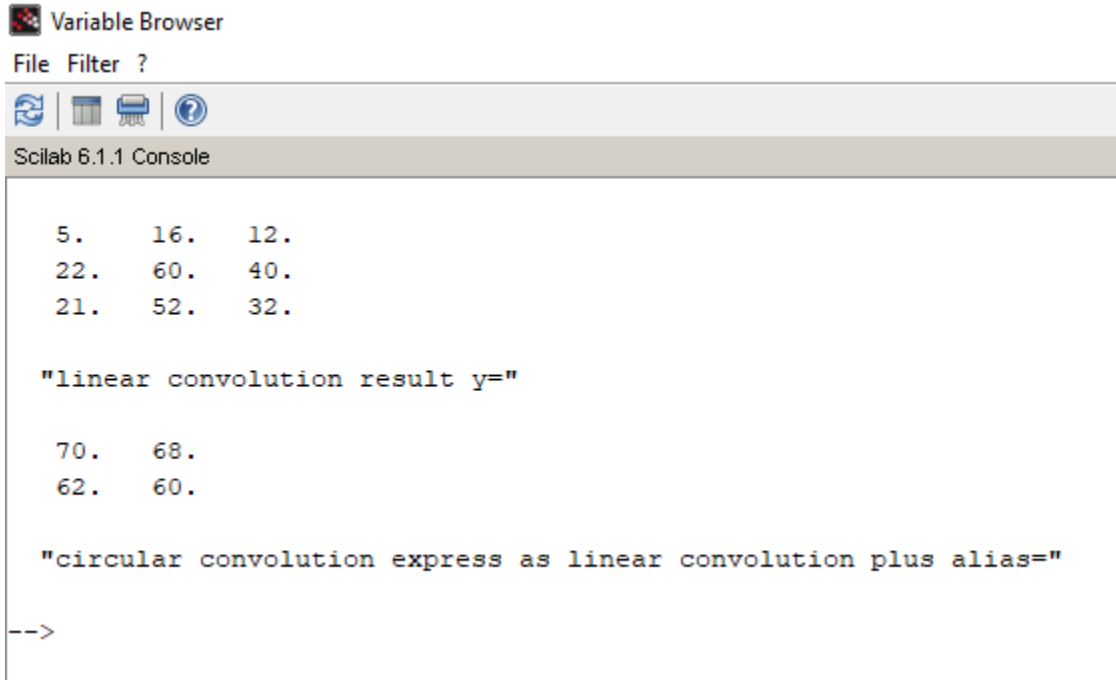
sources code:

```

clc;
x=[1,2;3,4];
h=[5,6;7,8];
y= conv2(x,h);
y1=[y(:,1)+y(:,4),y(:,2)];
y2=[y1(1,:)+y1(4,:);y1(2,:)];
disp(y,'linear convolution result y=');
disp(y2,'circular convolution express as linear
convolution plus alias=');

```

OUTPUT:



The image shows a screenshot of the Scilab Variable Browser window. The window has a title bar 'Variable Browser' and a menu bar 'File Filter ?'. Below the menu bar is a toolbar with icons for refresh, save, print, and help. The main area is titled 'Scilab 6.1.1 Console' and displays the following output:

```
5.    16.    12.
22.    60.    40.
21.    52.    32.

"linear convolution result y="

70.    68.
62.    60.

"circular convolution express as linear convolution plus alias="

-->
```

PRACTICAL NO : 3

AIM :- 'Linear Cross Correlation of 2D Martix' , "Circular Correlation Between two signal" and ' Linear auto correlation of 2D Matrix '.

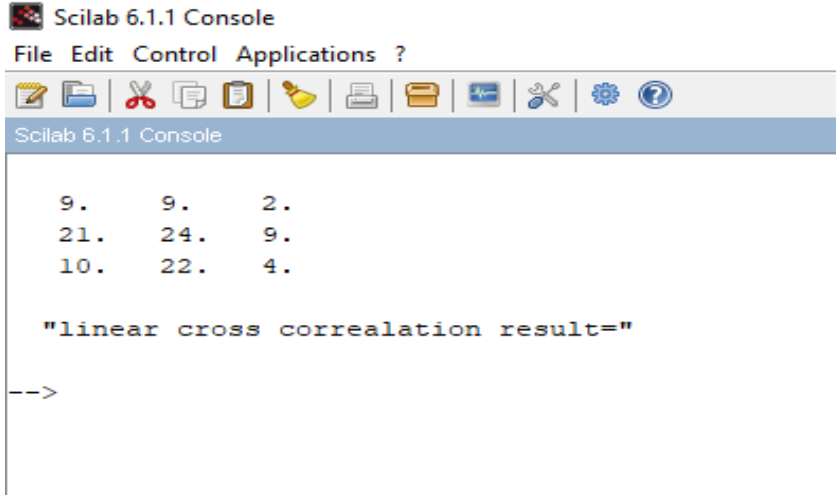
a) :- Linear Cross Correlation of 2D Martix.

sources code:

```
clc;
x=[3,1;2,4];
h1=[1,5;2,3];
h2=h1(:, $:-1:1);
h=h2($:-1:1, :);
y=conv2(x,h);
```

```
disp(y,"linear cross correlation result=");
```

OUTPUT:



```
Scilab 6.1.1 Console
File Edit Control Applications ?
[Icons]
Scilab 6.1.1 Console
  9.    9.    2.
 21.   24.   9.
 10.   22.   4.

"linear cross correlation result="
-->
```

b) :- Circular Correlation Between two signal .

sources code:

```
clc ;
```

```
x = [1 ,5;2 ,4];
```

```
h = [3 ,2;4 ,1];
```

```
h = h(:, $ : -1:1) ;
```

```
h = h($: -1:1 ,:);
```

```
X = fft2 (x);
```

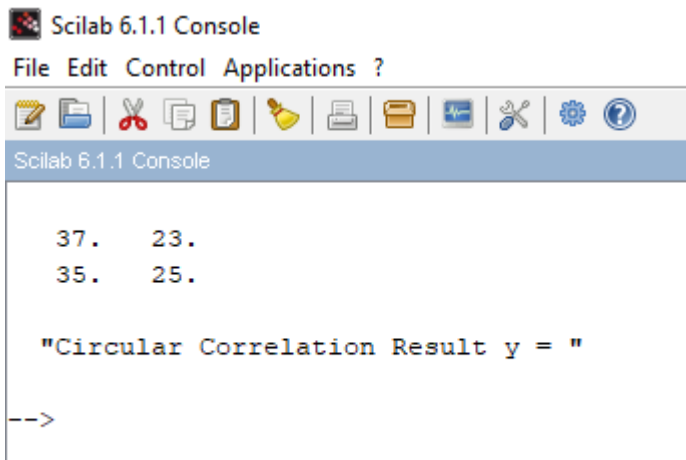
```
H = fft2 (h);
```

```
Y = X.*H;
```

```
y = ifft (Y);
```

```
disp(y, "Circular Correlation Result y = ");
```

OUTPUT:



```
Scilab 6.1.1 Console
File Edit Control Applications ?
[Icons]
Scilab 6.1.1 Console

37.    23.
35.    25.

"Circular Correlation Result y = "

-->
```

c) :- Linear auto correlation of 2D Matrix .

sources code:

`clc ;`

`x1 = [1 ,1;1 ,1];`

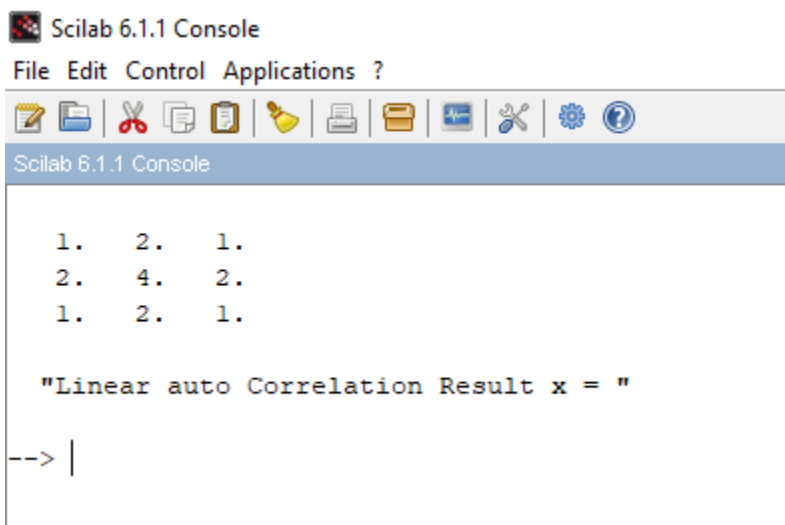
`x2 = x1 (:,$:-1:1);`

`x2 = x2($:-1:1 ,:);`

`x = conv2 (x1 ,x2);`

`disp(x, "Linear auto Correlation Result x = ");`

OUTPUT:-



```
Scilab 6.1.1 Console
File Edit Control Applications ?
[Icons]
Scilab 6.1.1 Console

1.    2.    1.
2.    4.    2.
1.    2.    1.

"Linear auto Correlation Result x = "

--> |
```

practical no 4

AIM: DFT of 4x4 gray scale image.

sources code:

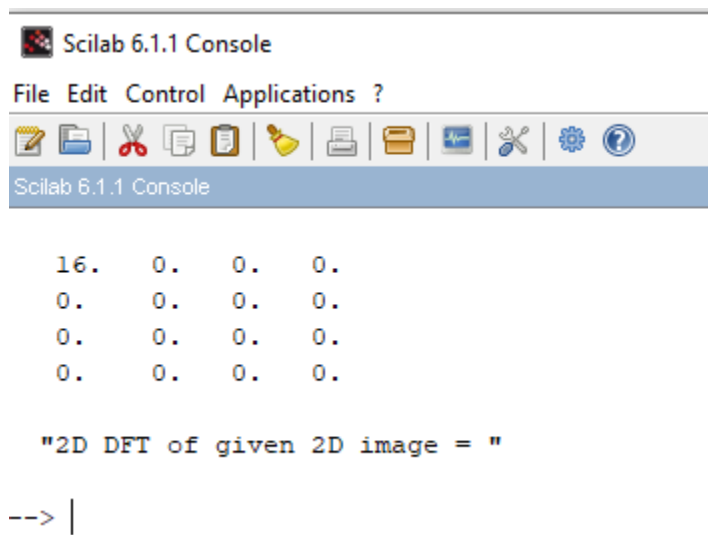
```
clc ;
```

```
f = [1,1,1,1;1,1,1,1;1,1,1,1;1,1,1,1];
```

```
t = fft2(f);
```

```
disp(t, "2D DFT of given 2D image = ");
```

OUTPUT:



The image shows a screenshot of the Scilab 6.1.1 Console window. The window has a title bar "Scilab 6.1.1 Console" and a menu bar with "File", "Edit", "Control", "Applications", and "?". Below the menu bar is a toolbar with various icons. The console area displays the output of the code: a 4x4 matrix of values and a string. The matrix is:

16.	0.	0.	0.
0.	0.	0.	0.
0.	0.	0.	0.
0.	0.	0.	0.

Below the matrix, the string "2D DFT of given 2D image = " is displayed. The prompt "--> |" is shown at the bottom of the console.

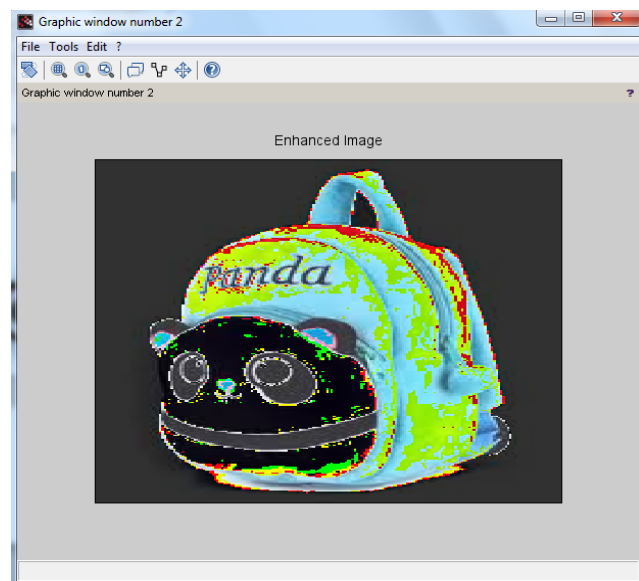
PRACTICAL NO : 6

AIM: Brightness enhancement of an image, Contrast Manipulation, image negative.

a) Brightness enhancement of an image.

sources code:

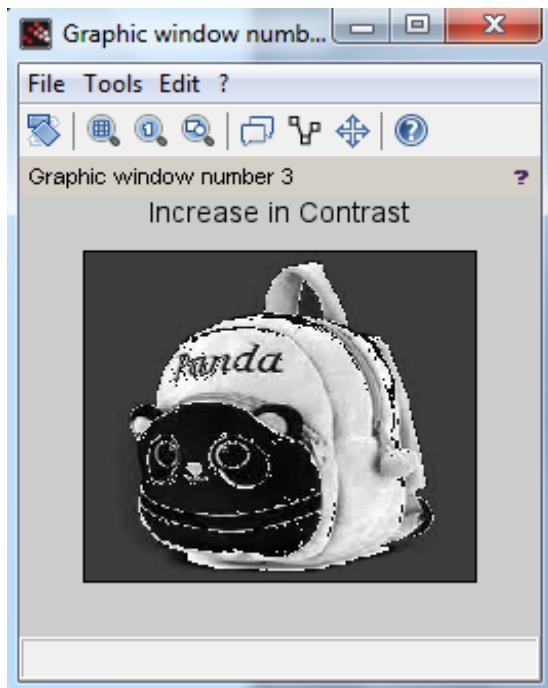
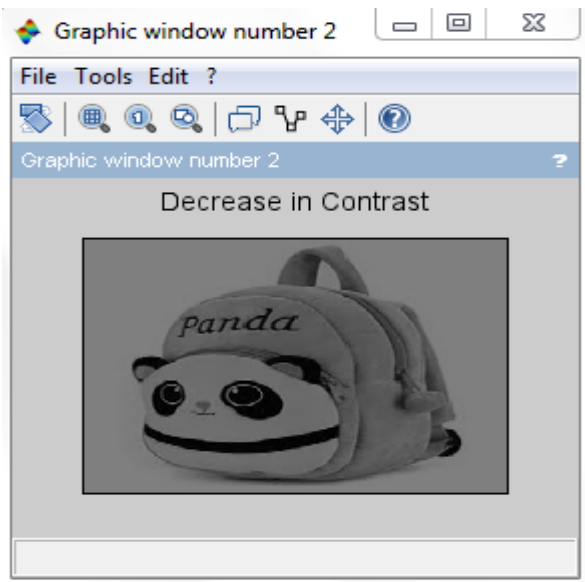
```
clc;  
a = imread ("C:\Users\mahendra\Documents\sandeep\school  
bag.jpg");  
b = double (a) +50;  
b = uint8 (b);  
figure (1)  
imshow (uint8(a));  
title ( ' Original Image ' )  
figure (2)  
imshow (uint8(b));  
title ( ' Enhanced Image ' ),  
output:
```



b) Contrast Manipulation

sources code:

```
clc ;  
close ;  
a = imread ("C:\Users\mahendra\Documents\sandeep\school  
bag.jpg");  
a = rgb2gray (a);  
b = double (a) *0.5;  
b = uint8 (b);  
c = double (b) *2.5;  
c = uint8 (c);  
figure (1)  
imshow(uint8(a));  
title ( 'Original Image' );  
figure (2)  
imshow(b);  
title ( 'Decrease in Contrast' );  
figure (3)  
imshow(c);  
title ( 'Increase in Contrast' );  
output:
```

c) image negative

sources code:

```
clc ;
```

```
close ;
```

```
a = imread ("C:\Users\mahendra\Documents\sandeep\school  
bag.jpg");
```

```
k = 255 - double (a);  
k = uint8 (k);  
figure(1);  
imshow (uint8(a));  
title ( 'Original Image' );  
figure(2);  
imshow (k);  
title ( 'Negative of Original Image' );  
output:
```



Practical no :7

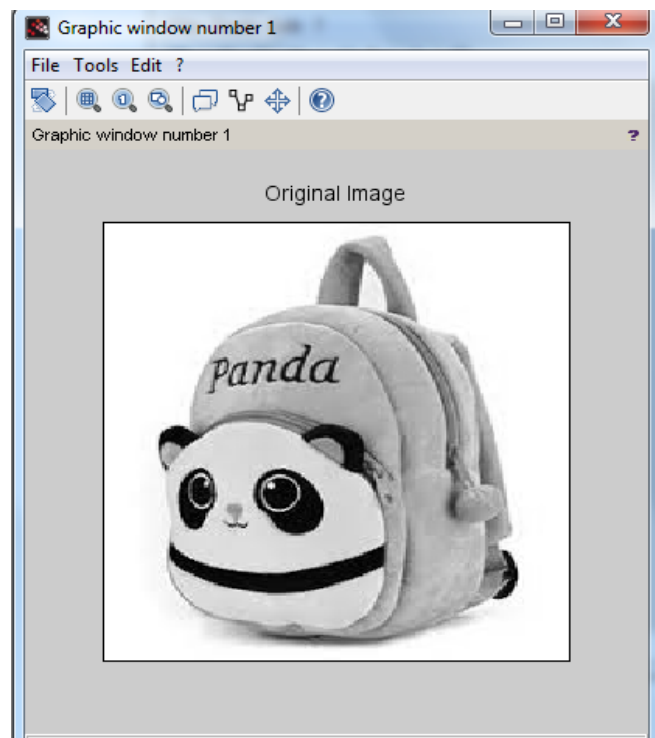
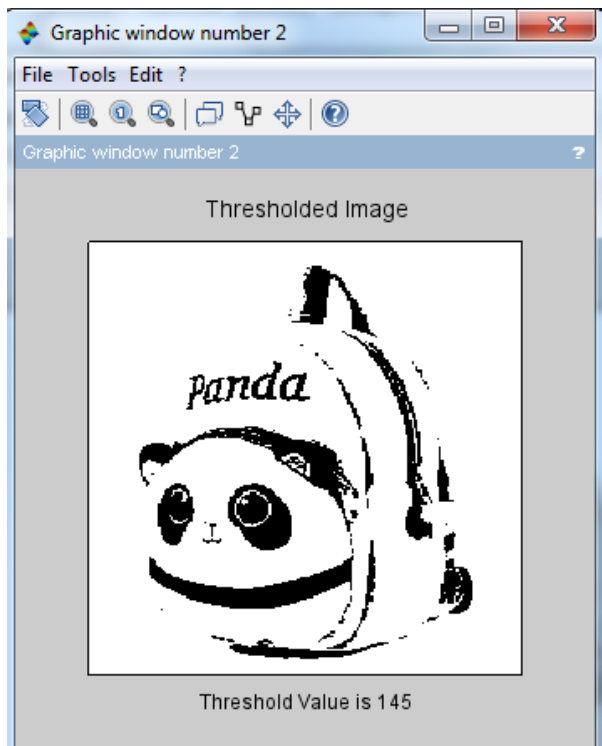
AIM : Perform threshold operation, perform gray level slicing without background.

a) Perform threshold operation

sources code:

```
clc ;
a = imread ('C:\Users\mahendra\Documents\sandeep\school
bag.jpg');
a = rgb2gray (a);
[m n] = size (a);
t = input("Enter the threshold parameter ");
for i = 1:m
for j = 1:n
if(a(i,j)<t)
b(i,j)=0;
else
b(i,j) =255;
end
end
end
figure (1)
imshow(uint8(a));
title ( ' Original Image ' )
figure (2)
imshow(uint8(b));
title ( ' Thresholded Image ' )
xlabel ( sprintf ( 'Threshold Value is %g ' ,t))
```

output:



b) perform gray level slicing without background.

```

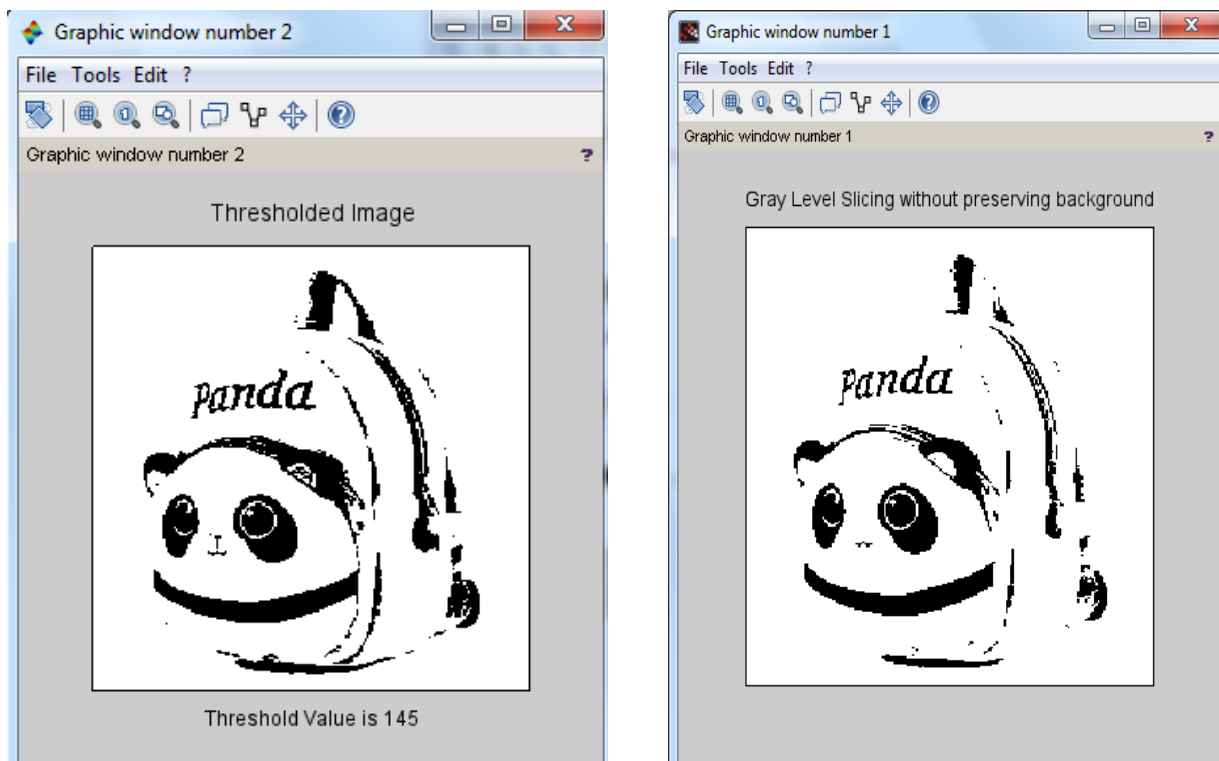
clc ;
x = imread("C:\Users\mahendra\Documents\sandeep\school bag.jpg");
x = rgb2gray(x);
y = double(x);
[m n] = size(y);
L = max(x);
a = round(L/2);
b = L;
for i = 1: m
    for j = 1: n
        if(y(i,j) >= a & y(i,j) <= b)
            z(i,j) = L;
        else
            z(i,j) = 0;
        end
    end
end

```

```

z = uint8 (z);
figure (1)
imshow(x)
title ( ' Original Image ' )
figure (2)
imshow(z);
title ( 'Gray Level Slicing without preserving background ' )

```



practical no 8

AIM : Image Segmentation (Differentiation of Gaussian function and Edge Detection)

a) Differentiation of Gaussian function

sources code:

```
sigma=input(' Enter the value of sigma : ')

i= -10:.1:10;

j= -10:.1:10;

r=sqrt(i.*i+j.*j);

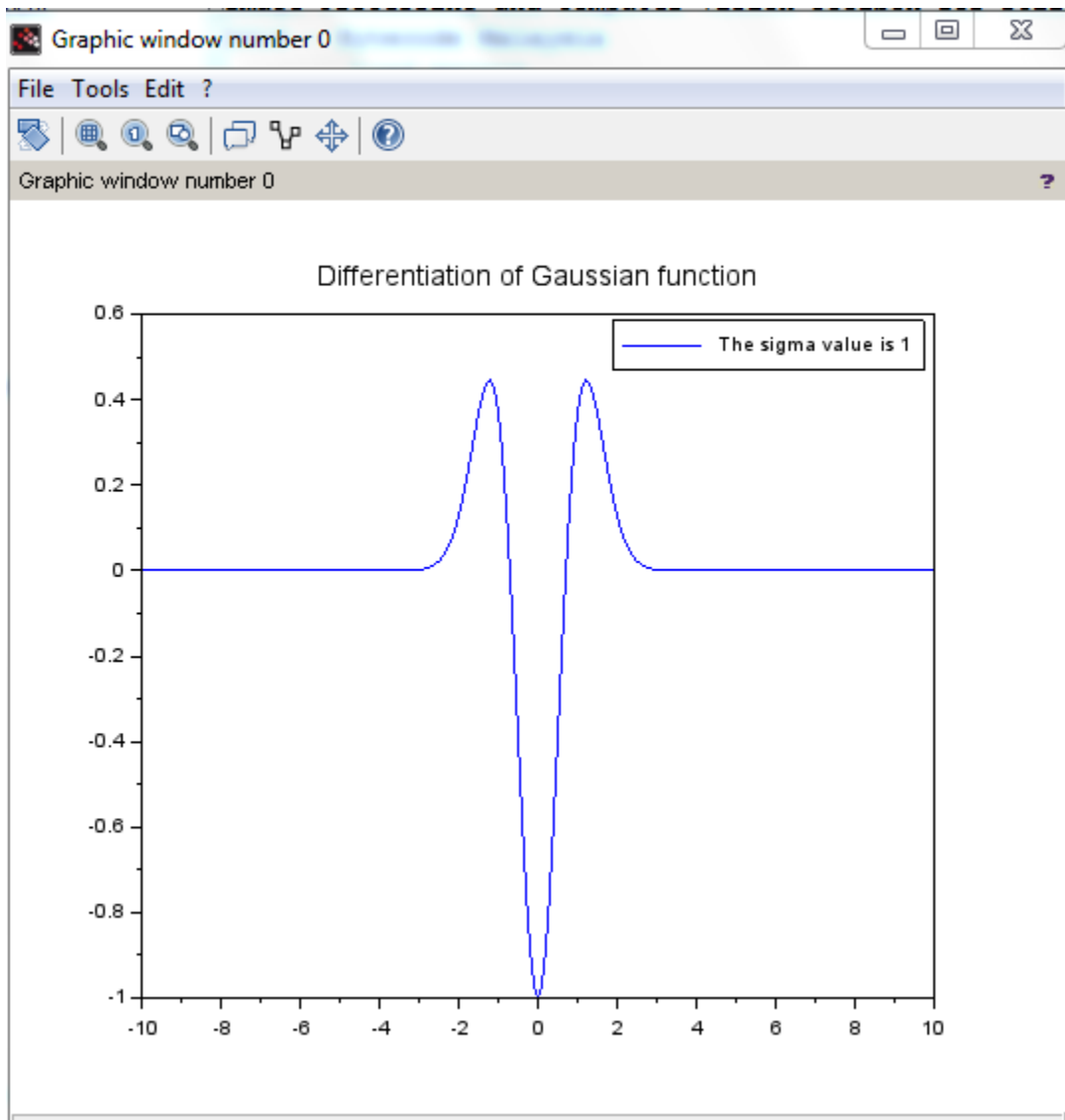
y=(1/( sigma ^2))*(((r.*r)/sigma ^2) -1).*exp(-r.*r/2*sigma ^2);

plot(i,y)

legend(sprintf(' The sigma value is %g ',sigma))

xlabel(' Differentiation of Gaussian function ')
```

output:



b) : Edge Detection

sources code:

img =

imread('C:\Users\COMP72\Documents\run\temple.jfif');

img=rgb2gray(img);

```
c=edge(img,'sobel',0.5)
```

```
d=edge(img,'prewitt')
```

```
e=edge(img,'canny')
```

```
f=edge(img,'log')
```

```
figure(1)
```

```
imshow(img)
```

```
title('Original Image')
```

```
figure(2)
```

```
imshow(c)
```

```
title('Sobel')
```

```
figure(3)
```

```
imshow(d)
```

```
title('Prewitt')
```

```
figure(4)
```

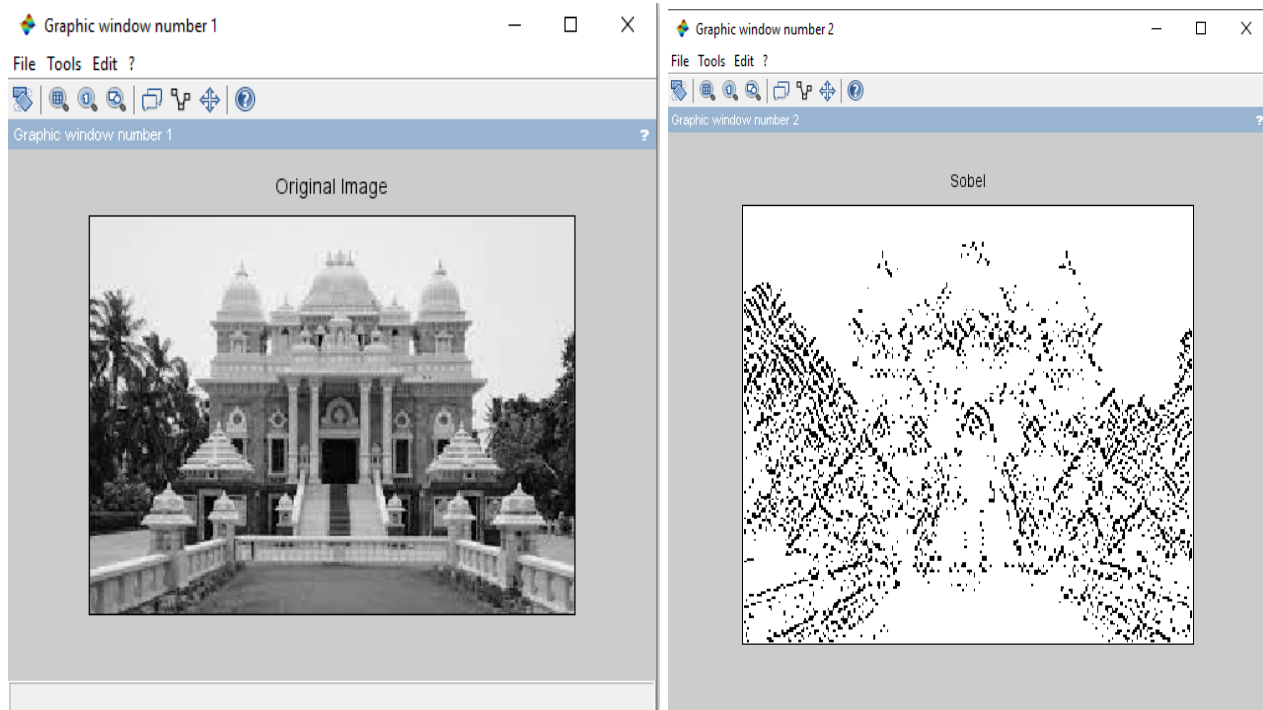
```
imshow(e)
```

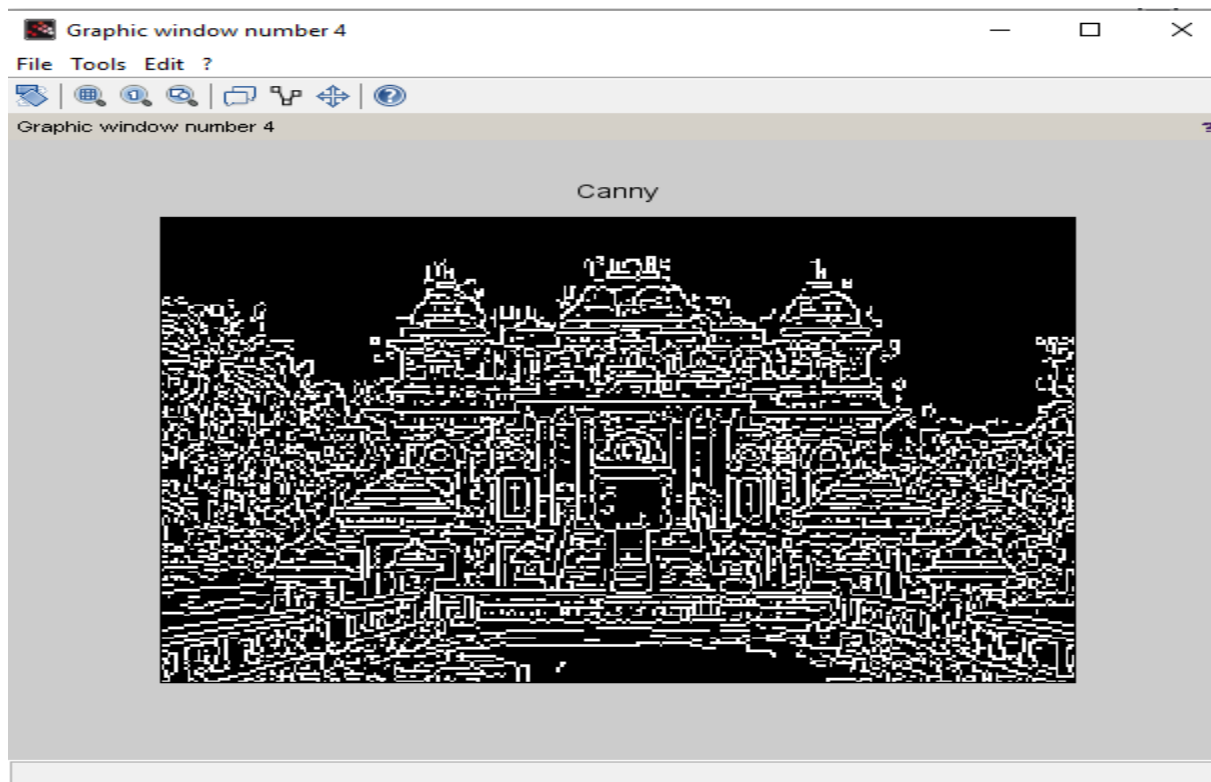
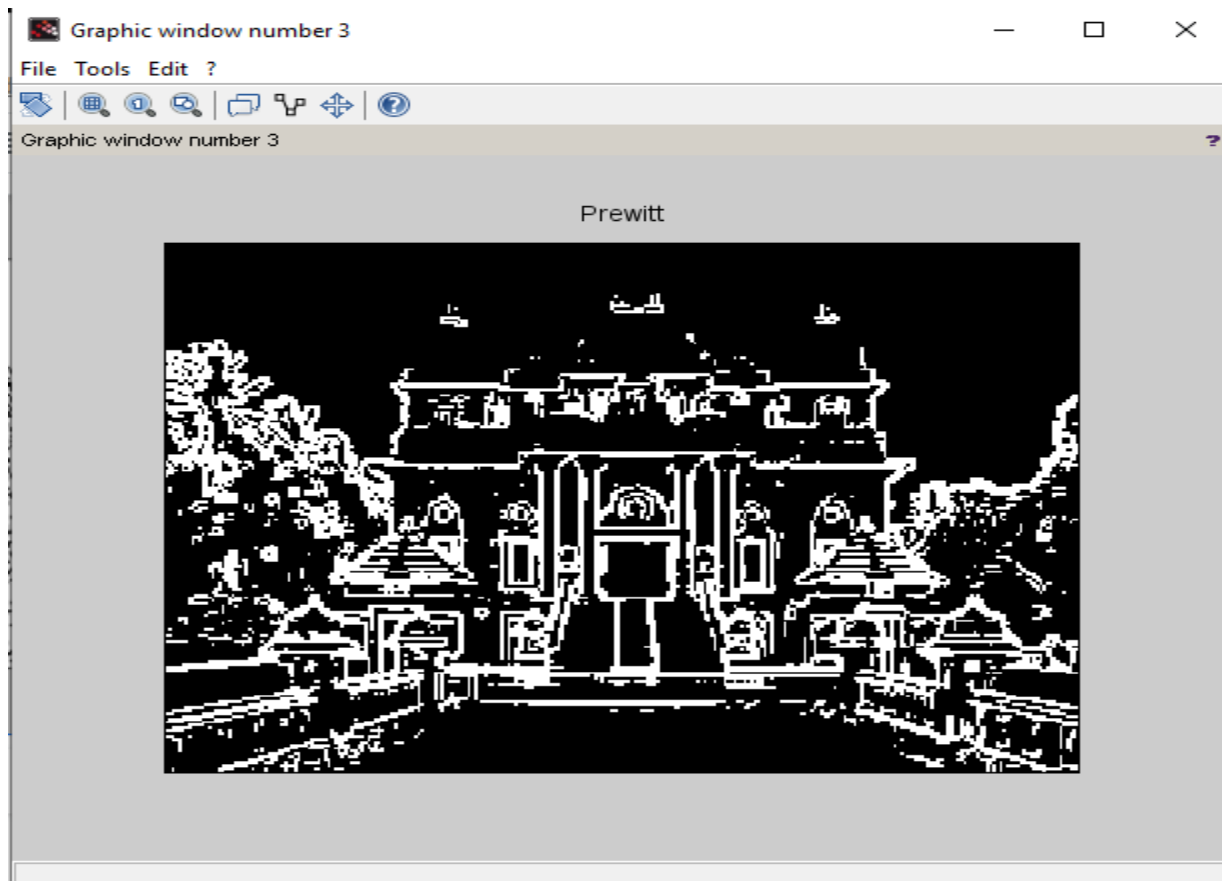
```
title('Canny')
```

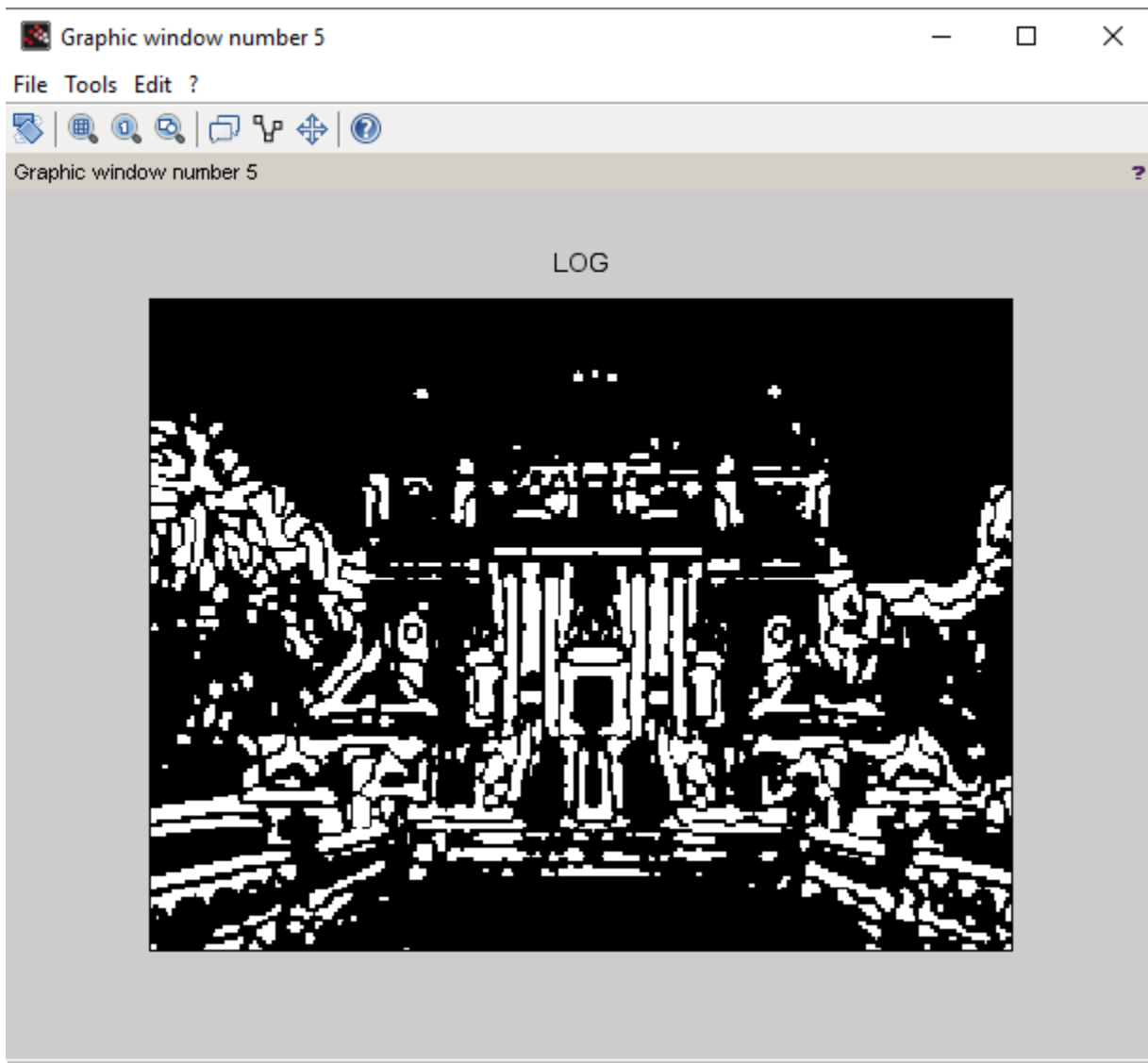
```
figure(5)
```


`imshow(f)`

`title('LOG')`







practical no :9

```
clc;
```

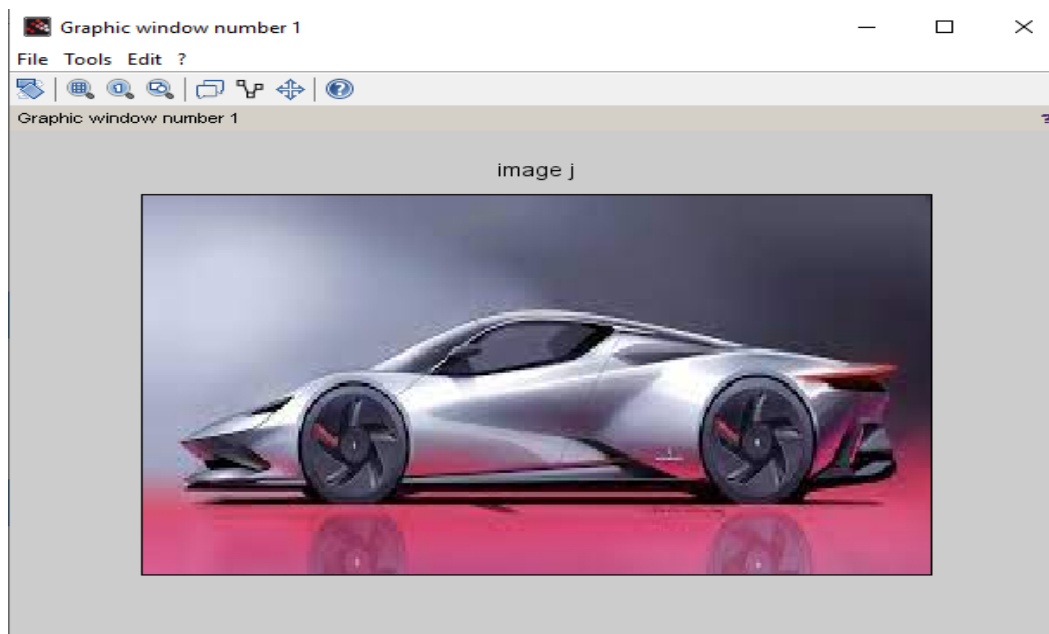
```
clear;
```

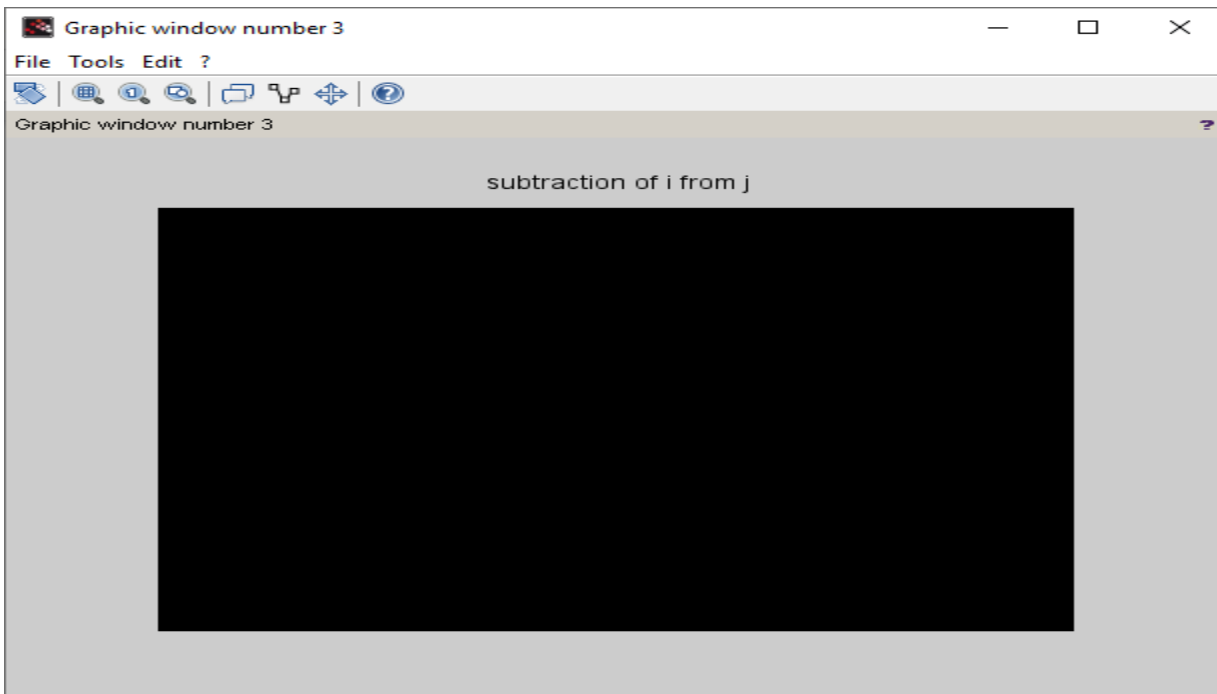
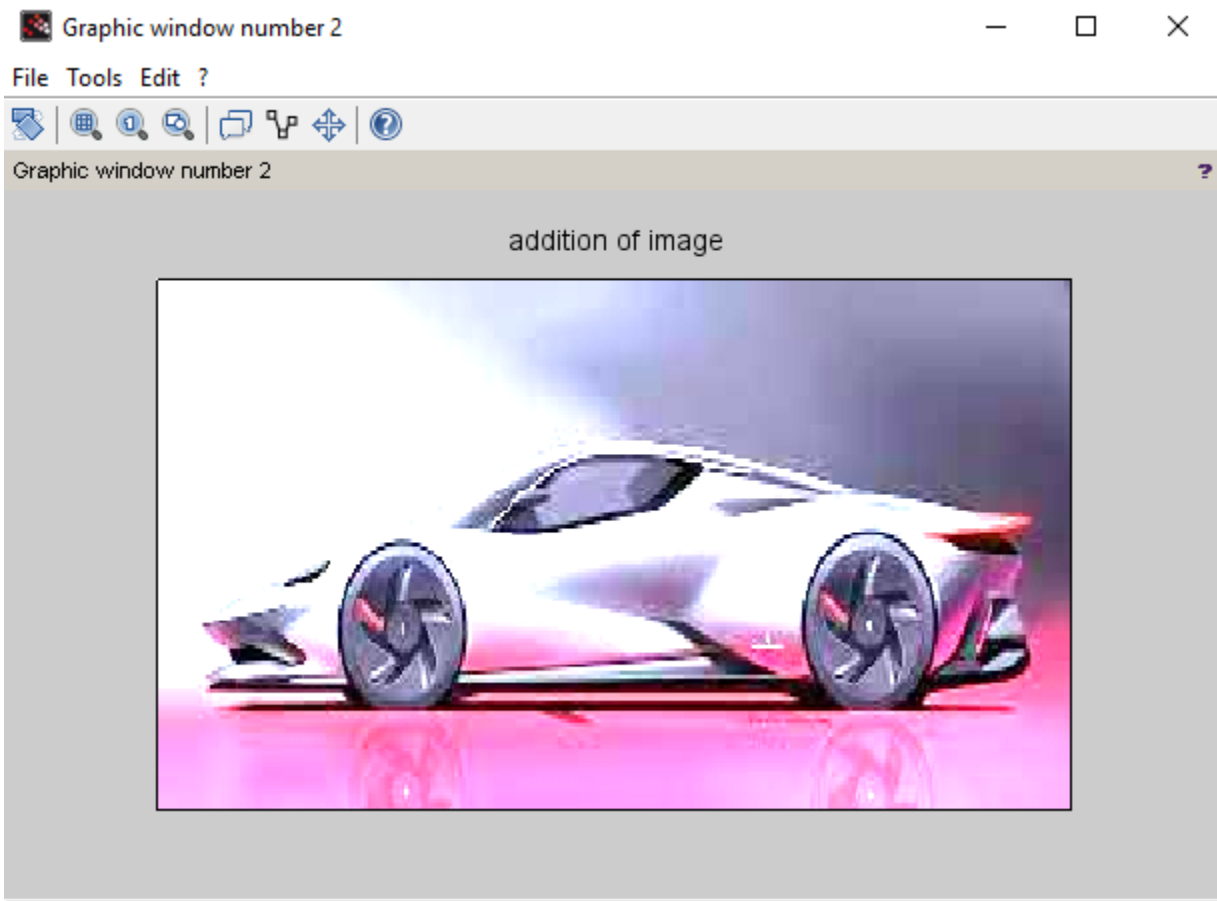
```
i=imread("C:\Users\COMP72\Documents\sandeep\car.jfif");
```

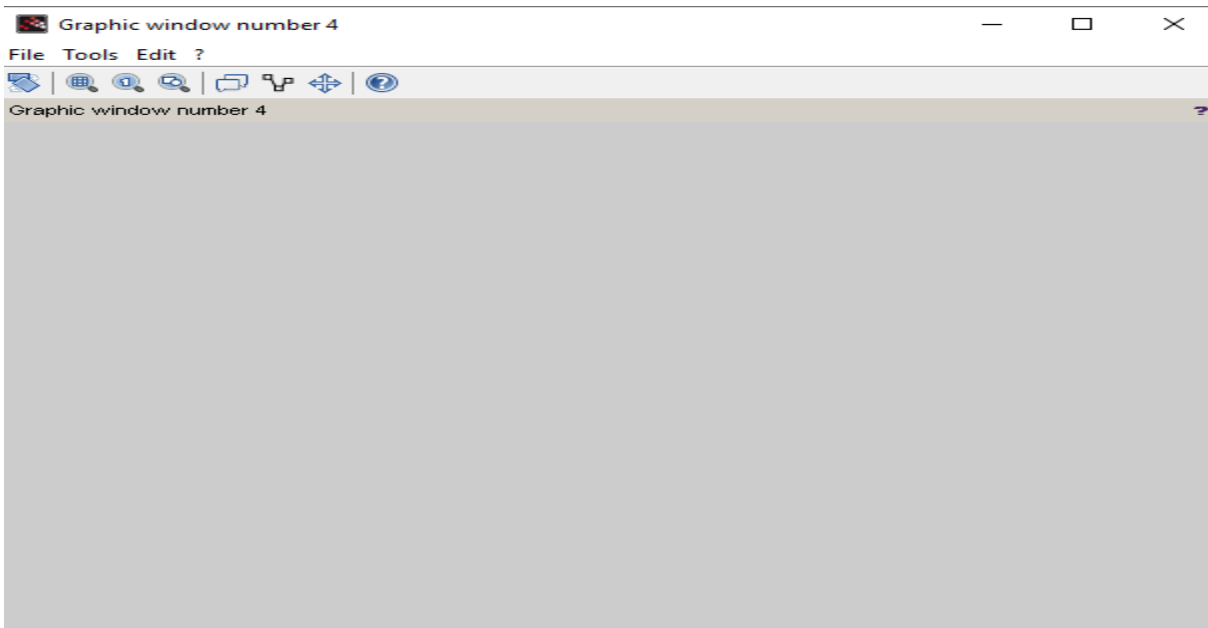
```
j=imread("C:\Users\COMP72\Documents\sandeep\car.jfif");
```

```
IMA=imadd(i,j);
```

```
figure(0);  
imshow(i);  
title("image i")  
  
figure(1);  
imshow(j);  
title("image j")  
  
figure(2);  
imshow(IMA);  
title("addition of image")  
  
IMS=imsubtract(j,i);  
  
figure(3);  
imshow(IMS);  
title("subtraction of i from j")  
  
IMS=imsubtract(i,j);  
  
figure(4);  
imshow(IMS1);  
title("subtraction of j from i");
```







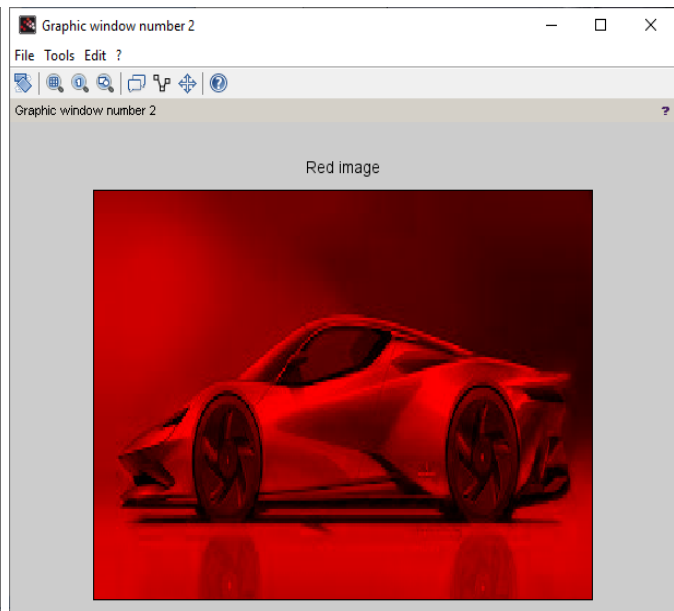
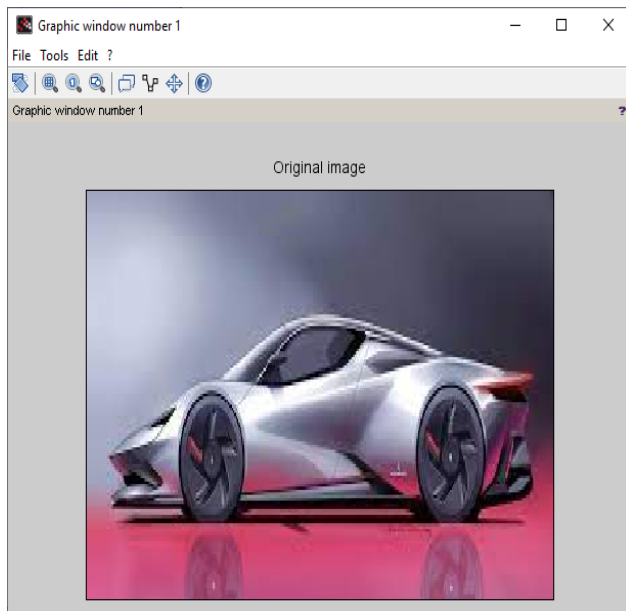
PRACTICAL NO :10

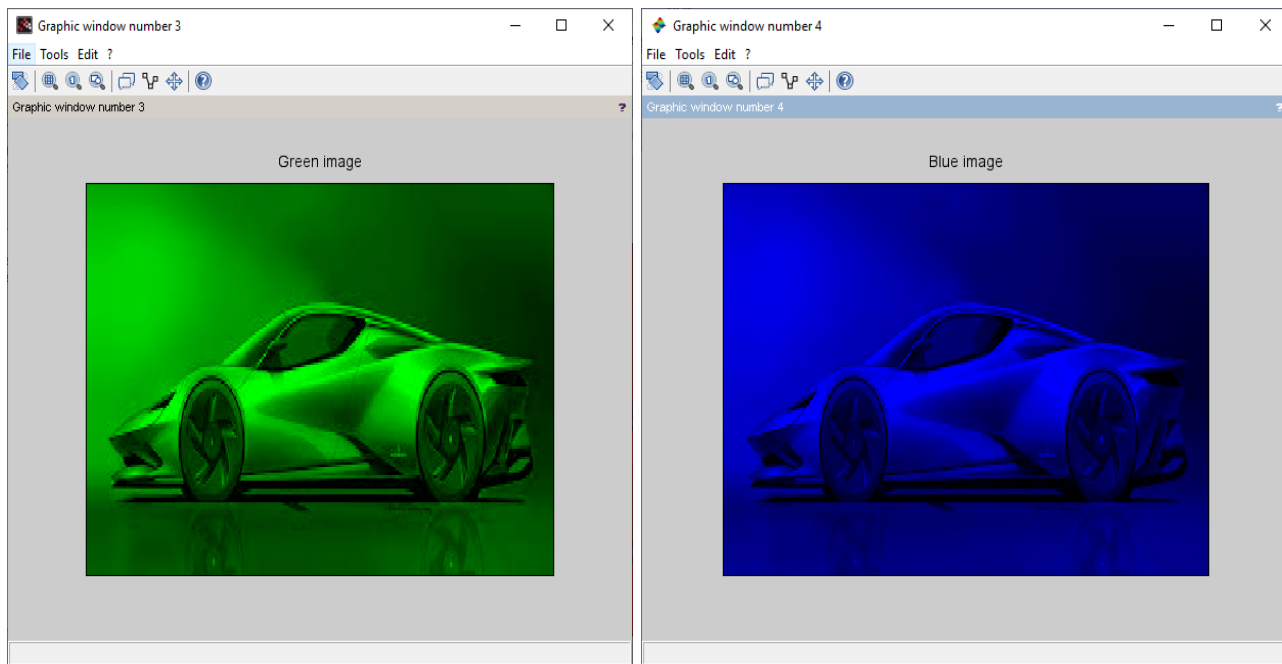
AIM:- COLOR IMAGE and BINARY IMAGE

a) COLOR IMAGE

```
clc;  
RGB =  
imread("C:\Users\COMP72\Documents\sandeep\car.jfif"  
")  
R = RGB  
G = RGB  
B = RGB  
R(:, :, 2) = 0;  
R(:, :, 3) = 0;  
G(:, :, 1) = 0;  
G(:, :, 3) = 0;  
B(:, :, 1) = 0;
```

```
B(:,:,2)=0;  
figure(1);  
imshow(RGB)  
title('Original image');  
figure(2);  
imshow(R)  
title('Red image');  
figure(3);  
imshow(G)  
title('Green image');  
figure(4);  
imshow(B)  
title('Blue image');  
OUTPUT:-
```





b) BINARY IMAGE :

```
clc;
```

```
a =
```

```
imread("C:\Users\COMP72\Documents\sandeep\car.jfif");
```

```
se = imcreate('rect', 3,3);
```

```
// Closing
```

```
b1 = imdilate(a, se);
```

```
b2 = imdilate(b1, se);
```

```
// Opening
```

```
a1 = imdilate(a, se);
```

```
a2 = imdilate(a1, se);
```

```
figure(1);
```

```
imshow(a);
```

```
title('Original image');
```

```
figure(2);  
imshow(a2);  
title('Open image');  
figure(3);  
imshow(b2);  
title('Close image');
```

OUTPUT:-

