GLA UNIVERSITY
ESTD. 2010
ऋते ज्ञानान्न मुक्तिः

GLA
UNIVERSITY
MATHURA
Established vide U.P. Act 21 of 2010.

Department of Computer Science & Application

Institute of Engineering & Technology

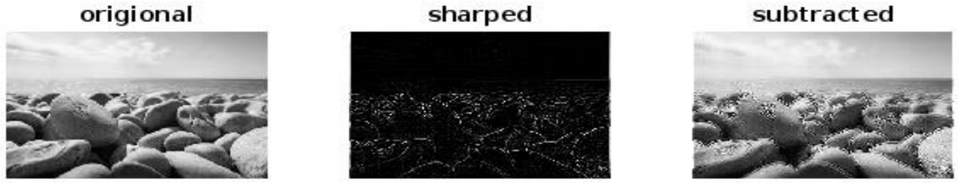
Submitted by:	**Submitted to:**
Name:- Abhishek pal	**Dr. Rajesh kumarTripathi**
Roll No:-201500013	
Sec-k(4)	

Q1.

OBJECTIVE:-Sharpen an image using Laplacian Filter.

Code:

```
%Laplacian
clear all
clc
i1 = imread('images.jpeg');
i1 = rgb2gray(i1);
[m n] = size(i1);
i1=double(i1);
f=[0 1 0 ; 1 -4 1 ; 0 1 0 ];
s=i1;
for i = 2:m-1
   for j = 2:n-1
       sum = 0;
       for k = 1:3
           for l = 1:3
               sum=sum+i1(i-2+k,j-2+l) *f(k,l);
           end
       end
       s(i,j) = sum;
   end
end
sm=i1+s;
figure(1),subplot(1,3,1),imshow(uint8(i1)),title('origional');
figure(1),subplot(1,3,2),imshow(uint8(s)),title('sharped');
figure(1),subplot(1,3,3),imshow(uint8(sm)),title('subtracted');
```



origional

sharped

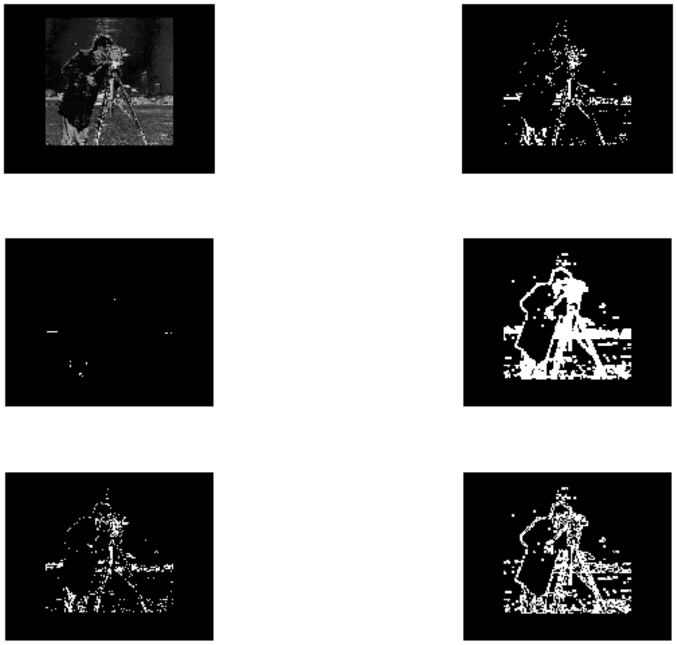
subtracted

Q2.

OBJECTIVE:-Perform Erode,Dilation,Opening&Closing.

Code:

```
clc
clear all
img= imread('ErodeGrayscaleImageWithRollingBallExample_01.png');
img1 = im2bw(img);
se = strel('disk',3);
img2= imerode(img1,se);
img3= imdilate(img1,se);
ib= img1-img2;
lb =img3-img1;
subplot(3,2,1),imshow(img);
subplot(3,2,2),imshow(img1);
subplot(3,2,3),imshow(img2);
subplot(3,2,4),imshow(img3);
subplot(3,2,5),imshow(ib);
subplot(3,2,6),imshow(lb);
```

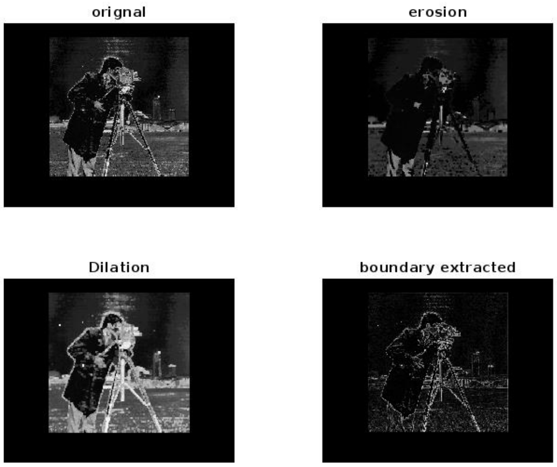


Q3

OBJECTIVE:-Perform Dilation erosion boundary extraction withooutusing direct function.

Code:

```
clear all
clc
a=imread('ErodeGrayscaleImageWithRollingBallExample_01.png');
p=size(a);
w=[1 1 1 ; 1 1 1 ; 1 1 1];
for x=2:1:p(1)-1
for y=2:1:p(2)-1
a1=[w(1)*a(x-1,y-1) ,
   w(2)*a(x-1,y),
   w(3)*a(x-1,y+1) ,
   w(4)*a(x,y-1) ,
   w(5)*a(x,y),
   w(6)*a(x,y+1),
   w(7)*a(x+1,y-1),
   w(8)*a(x+1,y),
   w(9)*a(x+1,y+1)];
A(x,y)=min(a1);%Erosion
B(x,y)=max(a1);%dilation
Sharp(x,y)=a(x,y)-A(x,y);
end
end
subplot(2,2,1),imshow(a),title('orignal');
subplot(2,2,2),imshow(A),title("erosion");
subplot(2,2,3),imshow(B),title("Dilation");
subplot(2,2,4),imshow(Sharp),title('boundary extracted');
```

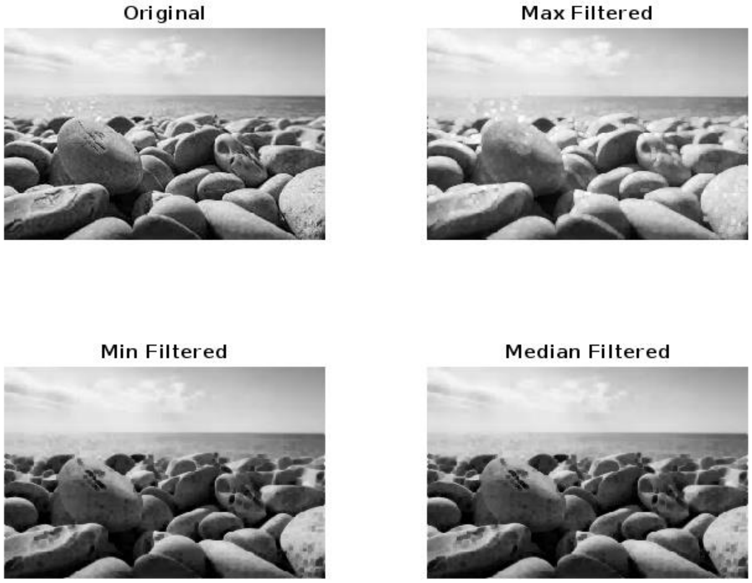

orignal
erosion
Dilation
boundary extracted

Q4.

OBJECTIVE:-Perform smoothing image using Max Min & Med filtering

Code:

```
i1 = imread('images.jpeg');
i1 = rgb2gray(i1);
[m n] = size(i1);
i1 = double(i1);
size = input('size of filter (odd number): ');
f = ones(size);
c = (size + 1)/2;
maxi = i1;
mini = i1;
medi = i1;
for i = c:m-c+1
   for j = c:n-c+1
       maxi(i,j) = max(max(i1(i-c+1:i-c+size,j-c+1:j-c+size)));
       mini(i,j) = min(min(i1(i-c+1:i-c+size,j-c+1:j-c+size)));
       medi(i,j) = median(median(i1(i-c+1:i-c+size,j-c+1:j-c+size)));
   end
end
figure(1), subplot(2,2,1), imshow(uint8(i1)), title('Original');
figure(1), subplot(2,2,2), imshow(uint8(maxi)), title('Max Filtered');
figure(1), subplot(2,2,3), imshow(uint8(mini)), title('Min Filtered');
figure(1), subplot(2,2,4), imshow(uint8(mini)), title('Median Filtered');
```

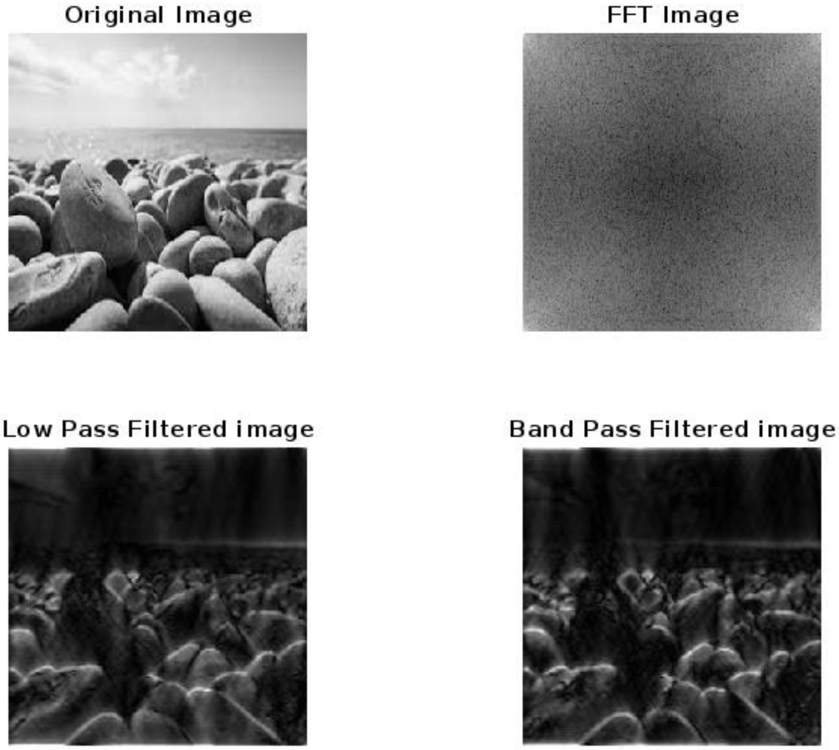

Original
Max Filtered
Min Filtered
Median Filtered

Q5.

OBJECTIVE:-Perform fast fourier Transform.

Code:

```
clear all
clc
mygrayimg = imread('images.jpeg');
mygrayimg = imresize(rgb2gray(mygrayimg),[256 256]);
subplot(2,2,1);
imshow(mygrayimg),title('Original Image');
myfftimage = fft2(mygrayimg);
tmp = abs(myfftimage);
mylogimg = log(1+tmp);
subplot(2,2,2);
imshow(mat2gray(mylogimg));
title('FFT Image');
[M,N]=size(myfftimage);
low=62;
band1=15;
band2=60;
mylowpassmask=ones(M,N);
mybandpassmask=ones(M,N);
for u = 1:M
   for v = 1:N
       tmp = ((u-(M+1))/2)^2+(v-(N+1)/2)^2;
       raddist = round((sqrt(tmp)));
       disp(raddist)
       if raddist > low
           mylowpassmask(u,v)=0;
       end
       if raddist > band2 || raddist < band1;
           mybandpassmask(u,v)=0;
       end
   end
end
f1 = fftshift(mylowpassmask);
f3 = fftshift(mybandpassmask);
resimage1 = myfftimage.*f1;
resimage3 = myfftimage.*f3;
r1 = abs(ifft2(resimage1));
subplot(2,2,3);
imshow(r1,[]),title('Low Pass Filtered image');
r3 = abs(ifft2(resimage3));
subplot(2,2,4);
imshow(r3,[]),title('Band Pass Filtered image');
```



Original Image

FFT Image

Low Pass Filtered image

Band Pass Filtered image

Q6

OBJECTIVE:-Perform smoothing using linear and order statistics filters.

Code:

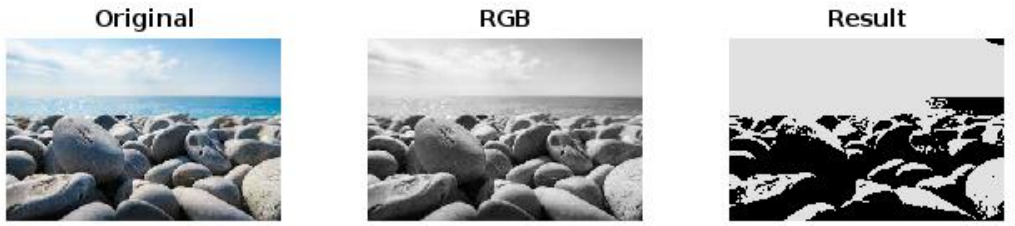
```
%Linear smoothing
clear all
clc
i1 = imread('images.jpeg');
i1 = rgb2gray(i1);
[m n] = size(i1);
i1 = double(i1);
size = input('size of filter (odd number): ');
f = ones(size);
c = (size + 1)/2;
i2 = i1;
for i = c:m-c+1
   for j = c:n-c+1
       sum = 0;
       for k = 1:size
           for l = 1:size
               sum = sum + i1(i-c+k, j-c+1)*f(k,1);
           end
       end
       i2(i,j) = sum / (size ^2);
   end
end
figure(1), subplot(1,2,1), imshow(uint8(i1));
figure(1), subplot(1,2,2), imshow(uint8(i1));
```

Q7

OBJECTIVE:- Perform Segmentation

Code:

```
Clear all
clc
img = imread('images.jpeg');
img1 = rgb2gray(img);
[m,n] = size(img1);
img2=img1;
img3=img1;
for i = 1 : m
   for j = 1 : n
       if(img1(i,j)<=150)
           img2(i,j)=0;
       else
           img2(i,j)=225;
       end
   end
end
figure(1), subplot(1,3,1), imshow(uint8(img)), title('Original');
figure(1), subplot(1,3,2), imshow(uint8(img1)), title('RGB');
figure(1), subplot(1,3,3), imshow(uint8(img2)), title('Result');
```

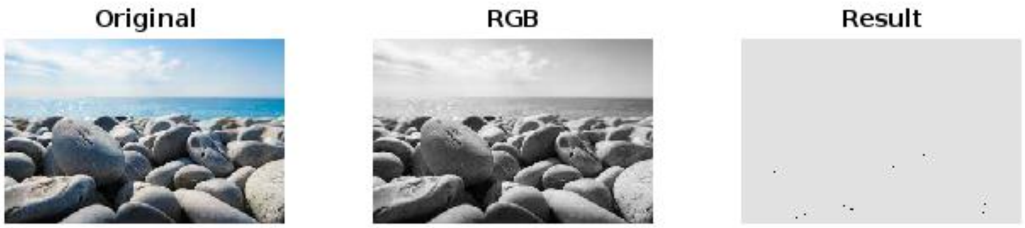


Q7

OBJECTIVE:-Perform Segmentation with average intensity

Code:

```
img1 = rgb2gray(img);
[m,n] = size(img1);
img2=img1;
s=0;
for a=1 :m
   for b=1:n
       s=s+img1(a,b);
   end
end
avg=s/(m*n);
for i = 1 : m
   for j = 1 : n
       if(img1(i,j)<=avg)
           img2(i,j)=0;
       else
           img2(i,j)=225;
       end
   end
end
figure(1), subplot(1,3,1), imshow(uint8(img)), title('Original');
figure(1), subplot(1,3,2), imshow(uint8(img1)), title('RGB');
figure(1), subplot(1,3,3), imshow(uint8(img2)), title('Result');
```



Original

RGB

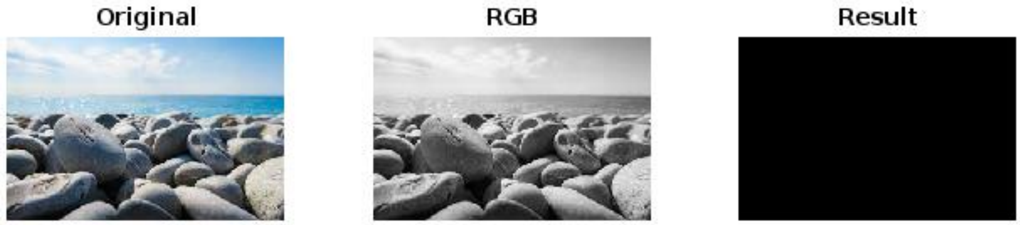
Result

Q8

OBJECTIVE:Perform Segmentation with Maximum intensity

Code:

```
img = imread('images.jpeg');
img1 = rgb2gray(img);
[m,n] = size(img1);
img2=img1;
max=0;
for a=1 :m
   for b=1:n
       if(max<img1(a,b))
           max=img1(a,b);
       end
   end
   end
for i = 1 : m
   for j = 1 : n
       if(img1(i,j)<=max)
           img2(i,j)=0;
       else
           img2(i,j)=225;
       end
   end
end
figure(1), subplot(1,3,1), imshow(uint8(img)), title('Original');
figure(1), subplot(1,3,2), imshow(uint8(img1)), title('RGB');
figure(1), subplot(1,3,3), imshow(uint8(img2)), title('Result');
```


Original
RGB
Result