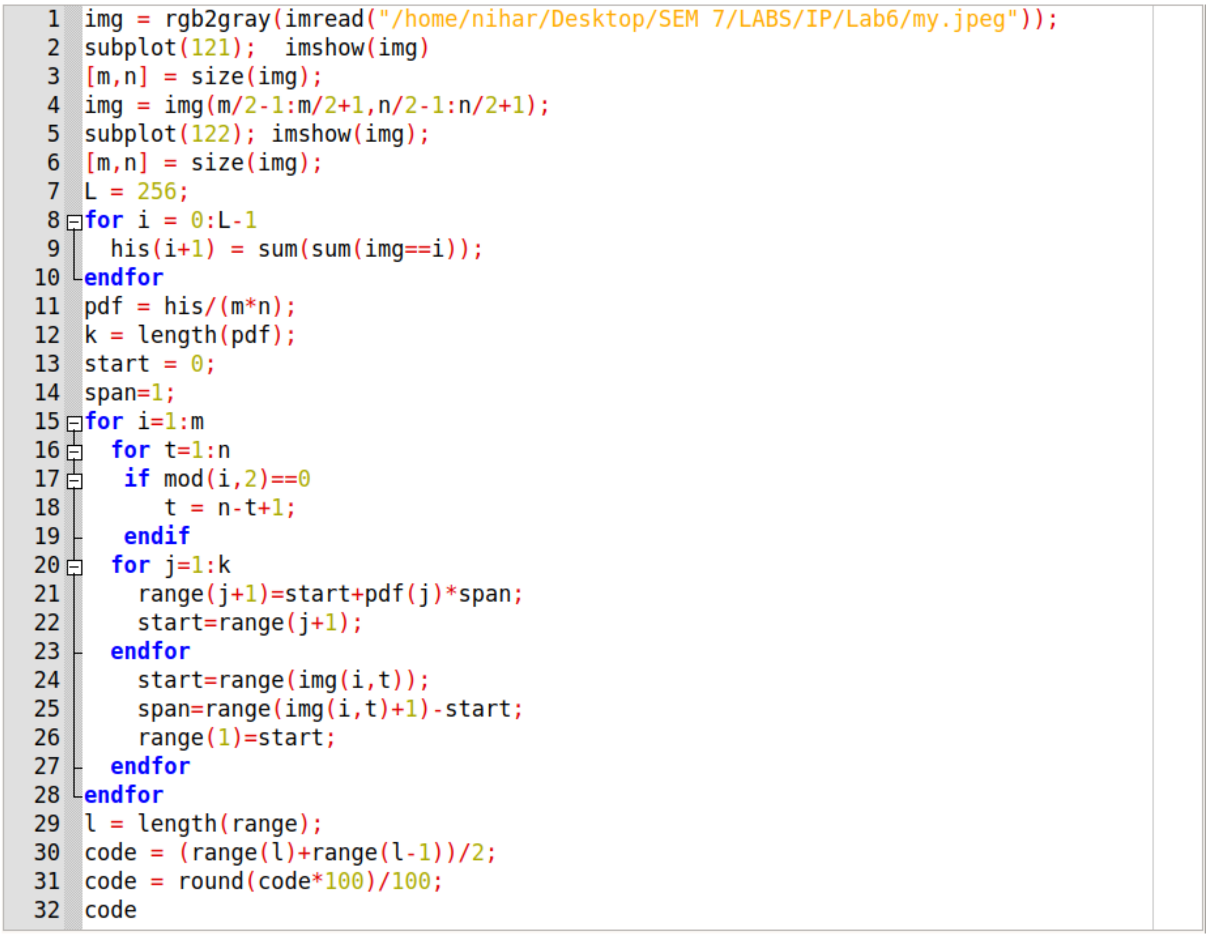
EXPERINMENT 12

Aim : Implementing Compression in Digital Image Processing.

❖ Exercises :

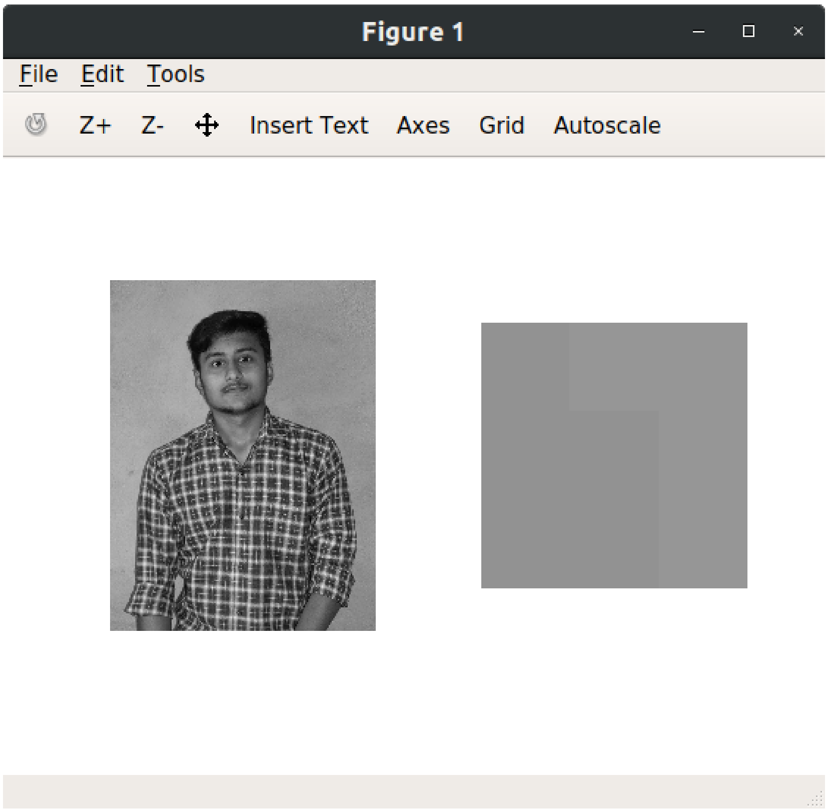
1. **Write an octave code to encode [arithmetic encoding] a 3x3 segment of your grayscale photo. Consider 3x3 segment element as vector of length 9 by considering image elements in the pattern shown below.**

Code :



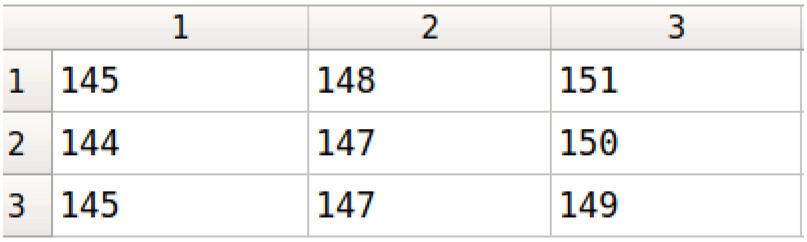
```
img = rgb2gray(imread("/home/nihar/Desktop/SEM 7/LABS/IP/Lab6/my.jpeg"));
subplot(121);  imshow(img)
[m,n] = size(img);
img = img(m/2-1:m/2+1,n/2-1:n/2+1);
subplot(122); imshow(img);
[m,n] = size(img);
L = 256;
for i = 0:L-1
  his(i+1) = sum(sum(img==i));
endfor
pdf = his/(m*n);
k = length(pdf);
start = 0;
span=1;
for i=1:m
  for t=1:n
   if mod(i,2)==0
      t = n-t+1;
   endif
  for j=1:k
    range(j+1)=start+pdf(j)*span;
    start=range(j+1);
  endfor
    start=range(img(i,t));
    span=range(img(i,t)+1)-start;
    range(1)=start;
  endfor
endfor
l = length(range);
code = (range(l)+range(l-1))/2;
code = round(code*100)/100;
code
```

Output :



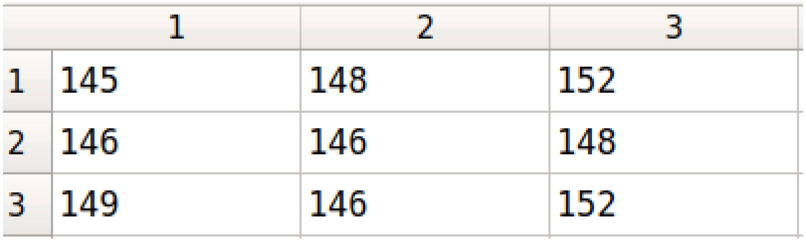
2. **Write an octave code to decode [arithmetic decoding] the image.**

- Pixels value before encoding



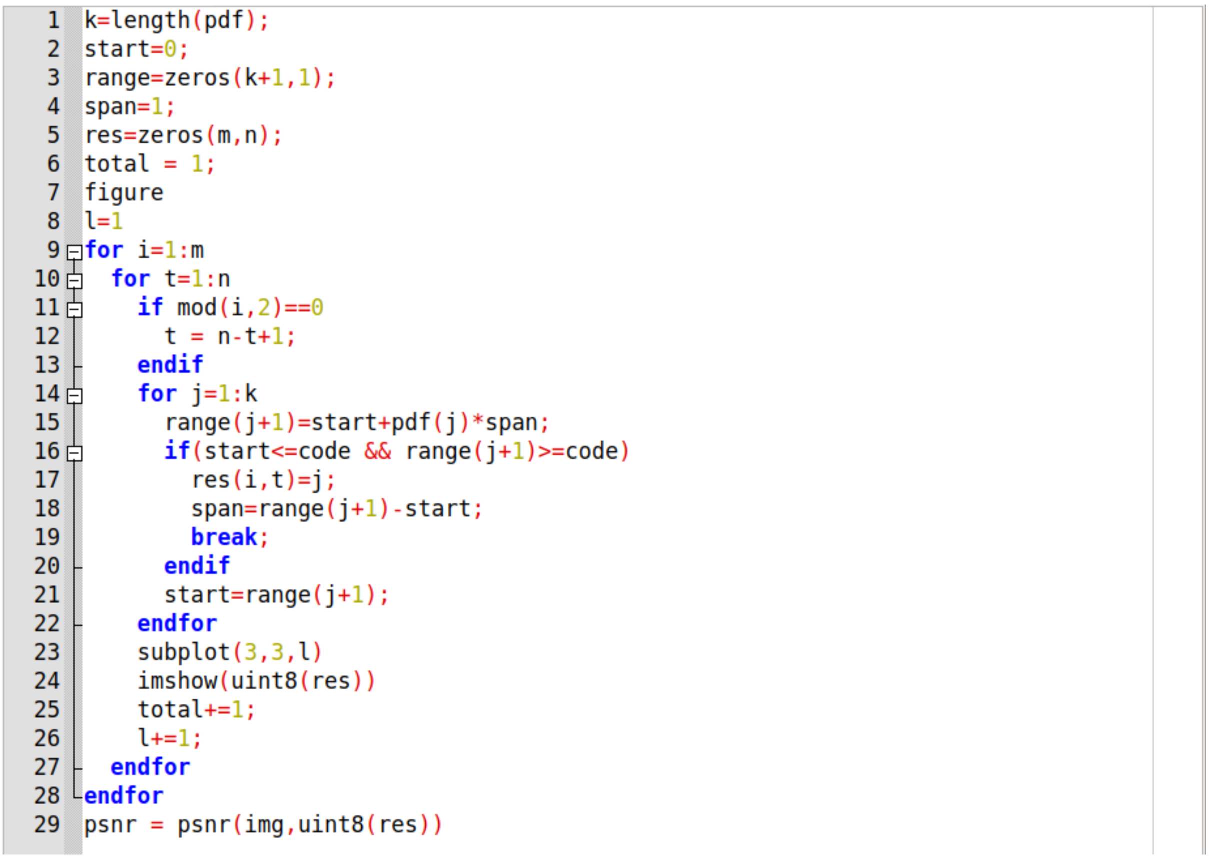
	1	2	3
1	145	148	151
2	144	147	150
3	145	147	149

- Pixels value after encoding



	1	2	3
1	145	148	152
2	146	146	148
3	149	146	152

Code :



```
k=length(pdf);
start=0;
range=zeros(k+1,1);
span=1;
res=zeros(m,n);
total = 1;
figure
l=1
for i=1:m
  for t=1:n
    if mod(i,2)==0
      t = n-t+1;
    endif
    for j=1:k
      range(j+1)=start+pdf(j)*span;
      if(start<=code && range(j+1)>=code)
        res(i,t)=j;
        span=range(j+1)-start;
        break;
      endif
      start=range(j+1);
    endfor
    subplot(3,3,l)
    imshow(uint8(res))
    total+=1;
    l+=1;
  endfor
endfor
psnr = psnr(img,uint8(res))
```

Output :

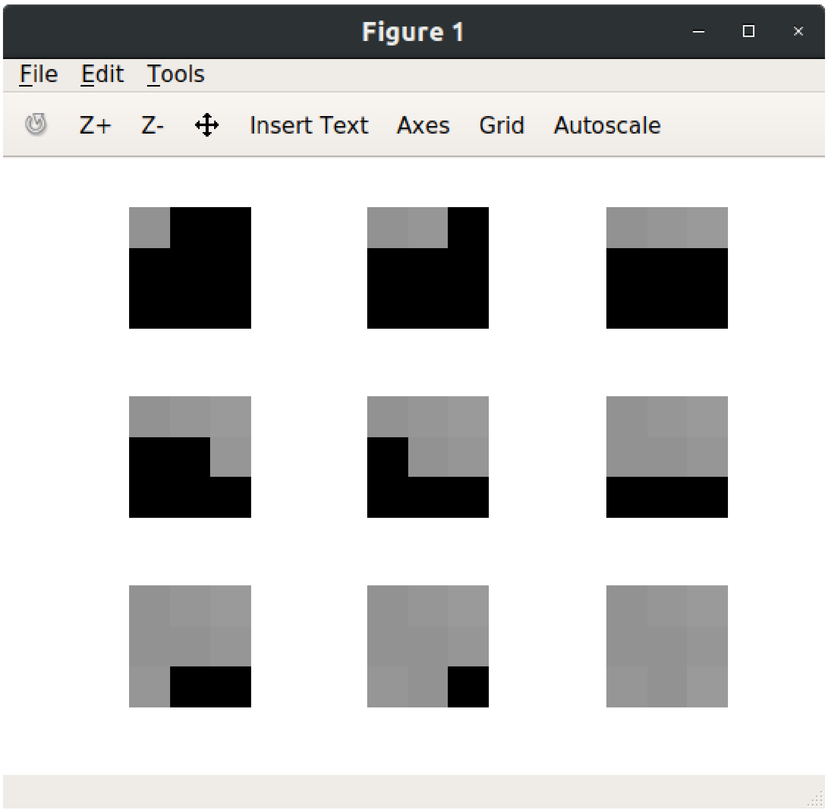


Figure 1
File
Edit
Tools
Z+
Z-
Insert Text
Axes
Grid
Autoscale