```
%Code to apply Shannon Fano coding to a grayscale image
clc;
clear all;
close all;

I=imread("Vidit.jpg");
if size(I,3)==3
    I=rgb2gray(I);
end
figure
imshow(I);
counts=imhist(I);             %Finding frequency of each gray level intensity.
p=counts/sum(counts);         %Normalizing histogram counts into
probabilities.

symbols=find(p>0)-1;          %Extracting only those intensity values that
appear.
p=p(p>0);                     %Removing all zero probability gray levels.

[p_sorted,idx]=sort(p,'descend');   %Sorting probabilities from highest to
lowest.
symbols_sorted=symbols(idx);        %Rearranging symbols in the same sorted
order.

codes=strings(1,length(symbols_sorted));
%Creating an empty string array to store Shannon–Fano binary codes.

codes=shannon_fano(symbols_sorted,p_sorted,codes,1,length(p_sorted));
%Calling the recursive function that generates Shannon–Fano codes.

disp("Top 20 Shannon–Fano Codes for Image Symbols:");
disp("GrayLevel   Probability     Code");
disp("-------------------------------------");

for i=1:min(20,length(symbols_sorted))
    fprintf("%3d         %.6f     %s\n", ...
        symbols_sorted(i),p_sorted(i),codes(i));
end
%Displaying only the most frequent gray levels and their corresponding codes.

Lavg=0;
for i=1:length(p_sorted)
    Lavg=Lavg+p_sorted(i)*strlength(codes(i));
end
%Computing the average code length using probability weighted sum.

H=0;
for i=1:length(p_sorted)
    H=H-p_sorted(i)*log2(p_sorted(i));
end
%Applying Shannon entropy formula H=-Σ(p*log2(p)) for binary coding.
```

```
disp("--------------------------------------");
fprintf("Entropy(H)=%.4f bits/pixel\n",H);
fprintf("AverageCodeLength(Lavg)=%.4f bits/pixel\n",Lavg);
fprintf("CodingEfficiency=%.2f %%\n",(H/Lavg)*100);
%Efficiency indicates how close coding is to the theoretical entropy limit.

function codes=shannon_fano(symbols,p,codes,startIdx,endIdx) %shannon fano
recursive function

    if startIdx>=endIdx
        return;
    end
    %Stopping recursion when only one symbol remains.

    totalProb=sum(p(startIdx:endIdx));
    %Calculating total probability of the current symbol group.

    runningSum=0;
    splitIdx=startIdx;

    for i=startIdx:endIdx
        runningSum=runningSum+p(i);
        %Finding cumulative probability until it reaches half of total.

        if runningSum>=totalProb/2
            splitIdx=i;
            break;
        end
    end

    for i=startIdx:splitIdx
        codes(i)=codes(i)+"0";
    end
    %Assigning binary 0 to the first probability subset.

    for i=splitIdx+1:endIdx
        codes(i)=codes(i)+"1";
    end
    %Assigning binary 1 to the second probability subset.

    codes=shannon_fano(symbols,p,codes,startIdx,splitIdx);
    codes=shannon_fano(symbols,p,codes,splitIdx+1,endIdx);
    %Recursively repeating the splitting until all symbols get a unique code.
end

Top 20 Shannon–Fano Codes for Image Symbols:
GrayLevel   Probability     Code
--------------------------------------
205         0.032039      000000
206         0.031840      000001
207         0.030322      00001
204         0.029353      00010
208         0.027358      00011
203         0.026625      001000
```

```
202          0.024656      001001
209          0.022472      00101
201          0.021647      00110
200          0.019520      00111
220          0.017276      010000
210          0.017230      010001
199          0.016375      010010
198          0.013766      010011
211          0.013000      0101000
197          0.011906      0101001
219          0.011014      010101
196          0.010806      010110
221          0.010477      010111
218          0.010008      0110000
-------------------------------------
Entropy(H)=7.2563 bits/pixel
AverageCodeLength(Lavg)=7.3458 bits/pixel
CodingEfficiency=98.78 %
```

Published with MATLAB® R2025b