1.
Function for gray-scale image:

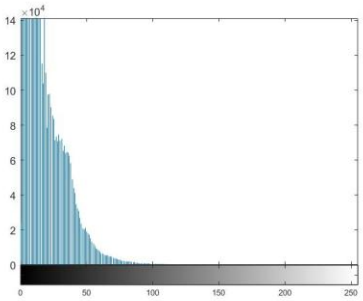
```
function [J] = myHistogramEq(I)
[r,c] = size(I);
X = zeros(1,256);
for k = 1:256
    for i = 1:r
        for j = 1:c
            if (k-1) == I(i,j)
                X(1,k) = X(1,k)+1;
            end
        end
    end
end
% X is the array which count the number of pixel of every
intensity value,using for loop too count the number.
% from 0 to 255.
X = X./(r*c); % Now it turns to pmf
S = zeros(size(X));
sum = 0;
col = 256;

for i=1:col
    sum = sum + X(1,i);
    S(1,i) = sum;
end
% S becomes cdf
S = round(S.*255);
L = zeros(size(I));
for i = 1:r
    for j = 1:c
            L(i,j) = S(1,I(i,j)+1);
    end
end
% do the actual mapping on the image
J = uint8(L);
% change double to 8bits unsigned int so that the image can
be used with imhist()
end
```

Image IMG_8787:
Script:

```
I = imread("IMG_8787.tiff");
```

```
% grey scale
J = rgb2gray(I);
K = myHistogramEq(J);
figure,imshow(J),axis off,
figure,imshow(K),axis off,
origin:
```

Transformed:



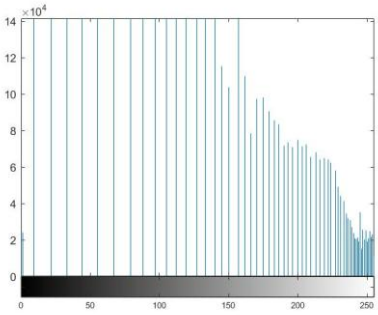
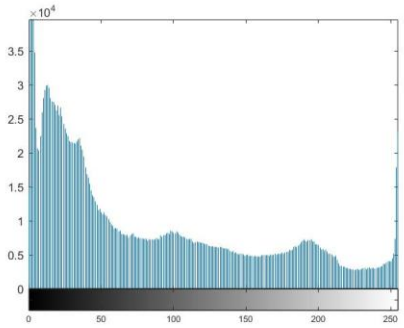


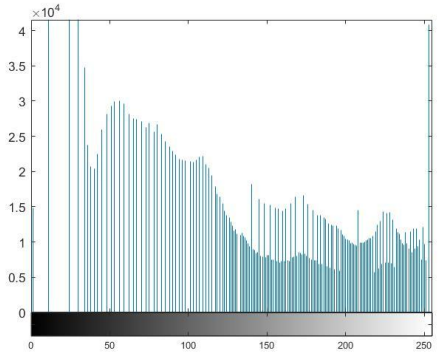
Image IMG_0798:
Script:

```
I = imread("IMG_0798.tiff");
% grey scale
J = rgb2gray(I);
K = myHistogramEq(J);
figure,imshow(J),axis off,
figure,imshow(K),axis off,
```

origin:

Transformed:





2.

Function for color image:

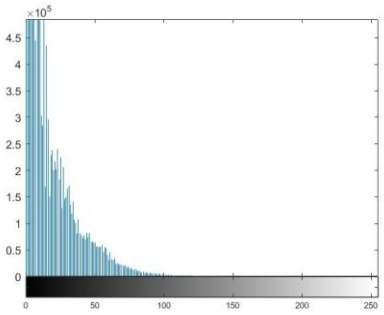
```
function [J] = MyColorHistEQ(I)
R = I(:,:,1);
G = I(:,:,2);
B = I(:,:,3);
%saparate the original image with red, green, and blue
```

```
components.
R_T = myHistogramEq(R);
G_T = myHistogramEq(G);
B_T = myHistogramEq(B);
% do equalization individually for three components
I(:,:,1) = R_T;
I(:,:,2) = G_T;
I(:,:,3) = B_T;
%comibine the three components together to get the new
image.
J = I;
end
```

Image IMG_8787:
Script:

```
I = imread("IMG_8787.tiff");
A = MyColorHistEQ(I);
figure,imshow(I),
figure,imshow(A),
origin:
```

Tranformed:



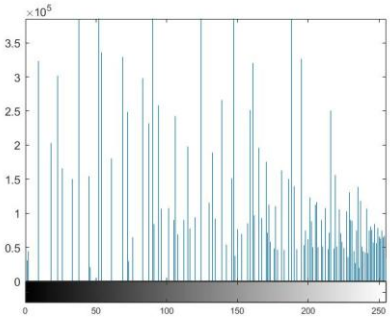


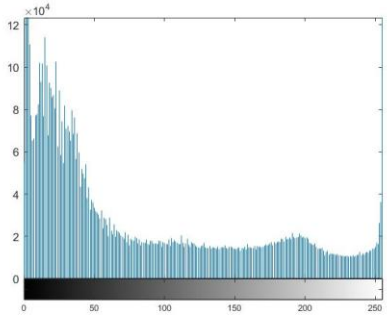
Image IMG_0798:
Script:

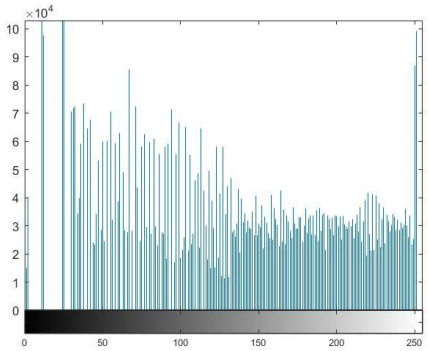
```
I = imread("IMG_0798.tiff");
A = MyColorHistEQ(I);
figure,imshow(I),
figure,imshow(A),
origin:
```


200
400
600
800
1000
1200
1400
1600
200
400
600
800
1000
1200
1400
1600

Tranformed:


200
400
600
800
1000
1200
1400
1600
200
400
600
800
1000
1200
1400
1600





Comment: It is a little surprised but reasonable that the histograms of the results after transfroming with the gray-scale image and after transforming individual color components are quite the same. I expect that the image with valleys should be brighter as the original one has too much pixels focued on the dark side. However, the result of the dining table is quite amazing that the image shows many colors after the tranformation, which is not I expected that it will just become brighter.