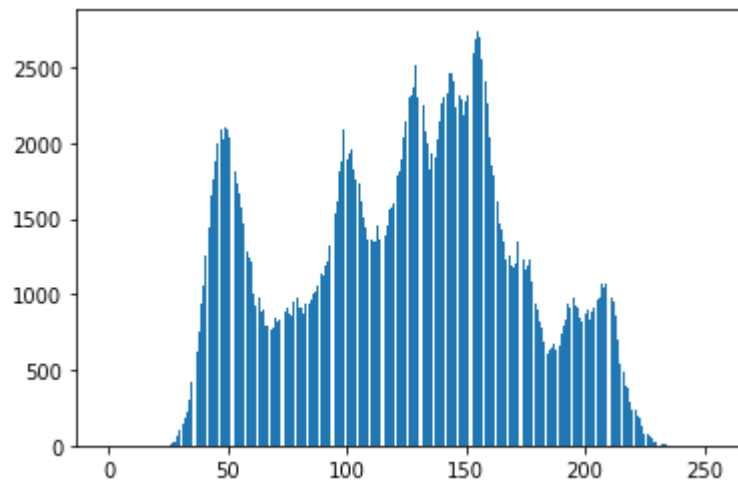
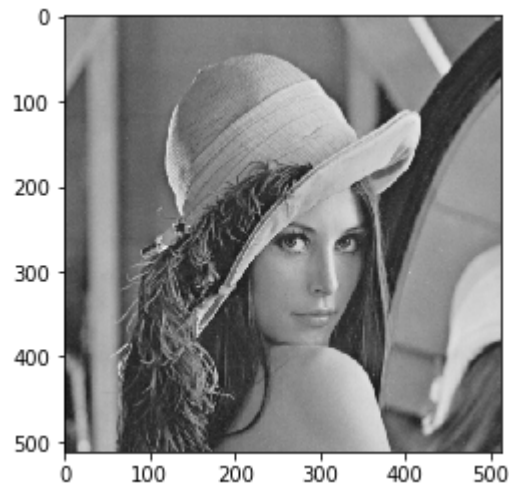


```
In [1]: import sys
from tqdm import tqdm_notebook
import numpy as np
import cv2
import math
import cv2
import matplotlib.pyplot as plt
import numpy as np
import sys
img = cv2.imread('lena.bmp', 0)
%matplotlib inline
```

原圖

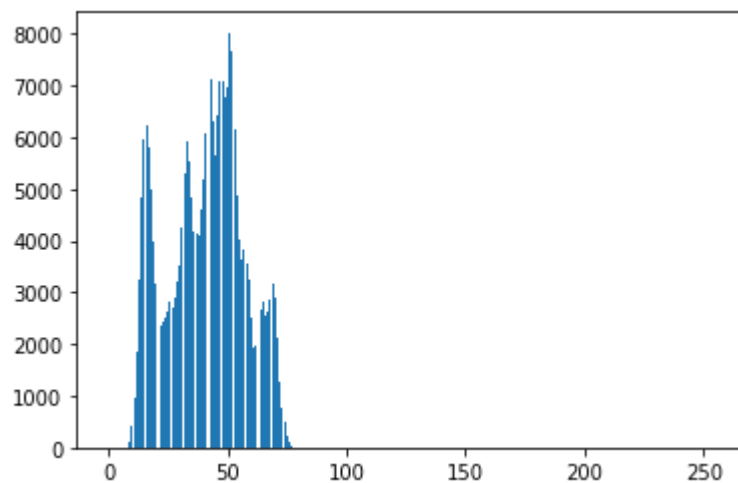
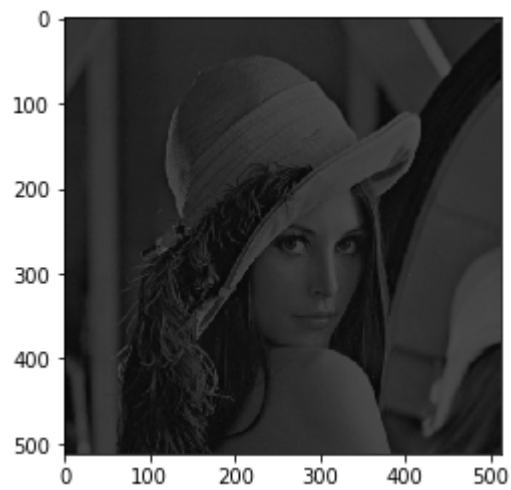
(a) original image and its histogram

```
In [3]: plt.imshow(img,cmap='gray', norm = None, vmin = 0, vmax = 0xff)
plt.show()
def histogram(image):
    hist = np.zeros(256,np.int)
    for i in image.reshape(-1,):
        hist[i]+=1
    return hist
hist=histogram(img)
plt.bar(range(len(hist)), hist)
plt.show()
```



(b) image with intensity divided by 3 and its histogram

```
In [21]: image=np.around((img/3), decimals=1).astype(np.int)
plt.imshow(image,cmap='gray', norm = None, vmin = 0, vmax = 0xff)
plt.show()
hist=histogram(image)
plt.bar(range(len(hist)), hist)
plt.show()
```



(c) image after applying histogram equalization to (b) and its histogram

```
In [22]: cdf=np.cumsum(hist)
eq=np.around(((cdf-cdf.min())/(cdf.max()-cdf.min()))*255).astype(np.int)
f = np.vectorize(lambda x:eq[x])
image=f(image)
plt.imshow(image,cmap='gray', norm = None, vmin = 0, vmax = 0xff)
hist=histogram(image)
plt.show()
plt.bar(range(len(hist)), hist)
plt.show()
```

