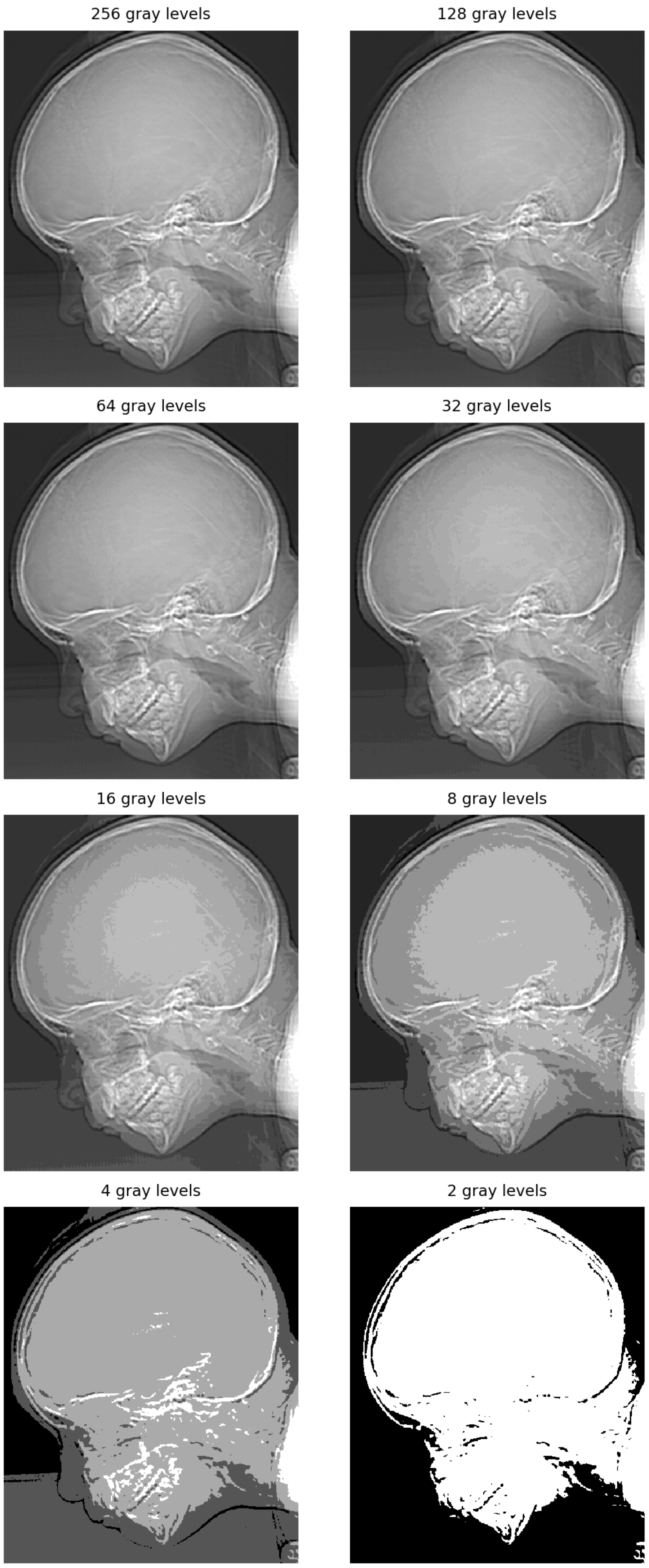
Duke Hong
Math155
504987610

Homework #1

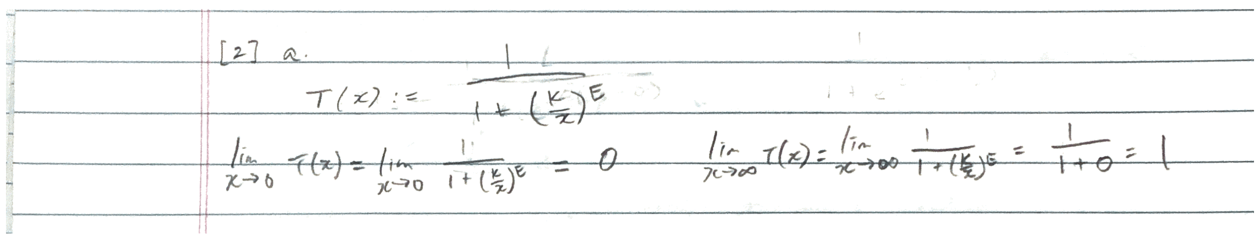
1a)

```
import numpy as np
from matplotlib import pyplot as plt
import math

# reduces intensity range from [0,255] to [0,val] using FLOOR
def reduceIntensity(img, val):
    # reduce to all black
    if val == 0:
        return img*0
    # out of range, return original image
    elif val > 256:
        return img
    else:
        return np.floor(img/(256/val))

if __name__ == '__main__':
    # read file image into numpy array
    img_orig = plt.imread('../images/Fig2.21(a).jpg', 0)

    # setup figure
    fig = plt.figure(figsize=(10,30))
    fig.suptitle('')
    plt.rcParams.update({'font.size': 7})

    # reduce orig_img by powers of 2 and plot resulting images
    for i in range(0,8):
        img = reduceIntensity(img_orig, 256/(2**i))
        ax = fig.add_subplot(8,2,i+1, frameon=False)
        ax.get_xaxis().set_visible(False)
        ax.get_yaxis().set_visible(False)
        ax.title.set_text("{} gray levels".format(int(256/(2**i))))
        ax = plt.imshow(img, cmap='gray')

    # spacing adjustment
    plt.subplots_adjust(wspace= -0.5, hspace=0.1)
    plt.show()
```

1b)


256 gray levels
128 gray levels
64 gray levels
32 gray levels
16 gray levels
8 gray levels
4 gray levels
2 gray levels

2a)

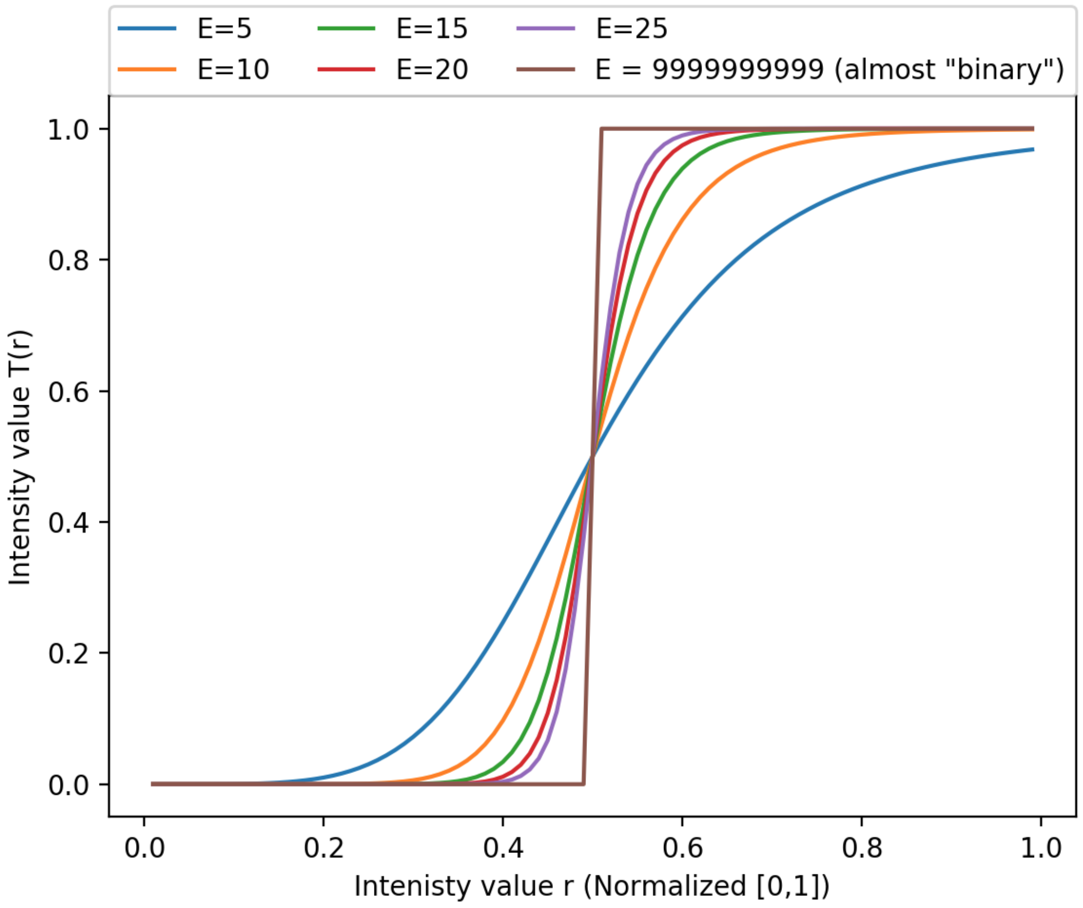


[2] a.

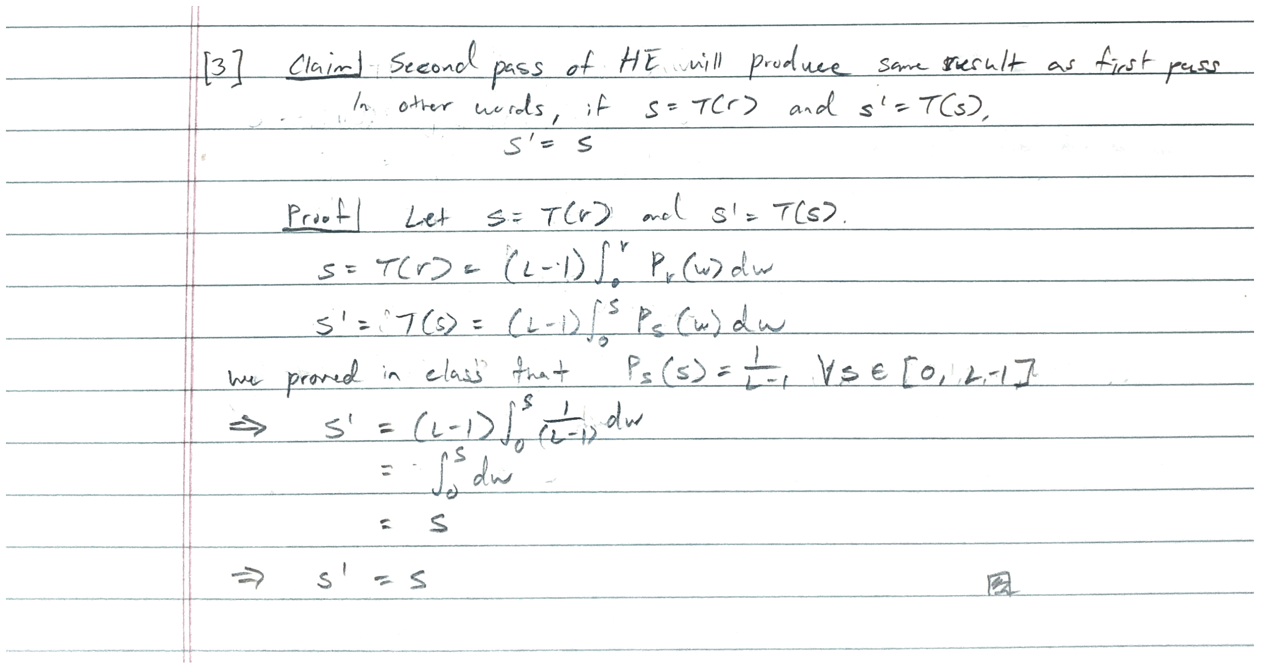
$$T(x) := \frac{1}{1+\left(\frac{K}{x}\right)^E}$$

$$\lim_{x\to 0} T(x) = \lim_{x\to 0} \frac{1}{1+\left(\frac{K}{x}\right)^E} = 0 \qquad \lim_{x\to \infty} T(x) = \lim_{x\to \infty} \frac{1}{1+\left(\frac{K}{x}\right)^E} = \frac{1}{1+0} = 1$$

2b)



3a)



[3] Claim: Second pass of HE will produce same result as first pass
In other words, if $s = T(r)$ and $s' = T(s)$,
$s' = s$

Proof: Let $s = T(r)$ and $s' = T(s)$.

$$s = T(r) = (L-1)\int_0^r P_r(w)\,dw$$

$$s' = T(s) = (L-1)\int_0^s P_s(w)\,dw$$

We proved in class that $P_s(s) = \frac{1}{L-1}, \ \forall s \in [0, L-1]$

$$\Rightarrow s' = (L-1)\int_0^s \frac{1}{(L-1)}\,dw$$

$$= \int_0^s dw$$

$$= s$$

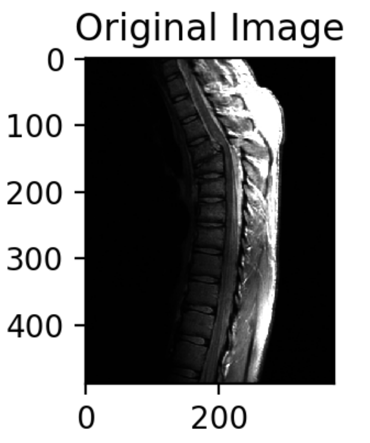
$$\Rightarrow s' = s \quad \blacksquare$$

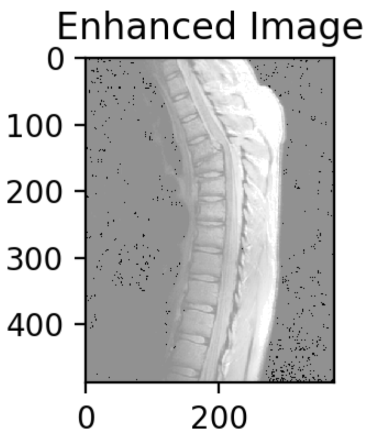
4a) and 4b)

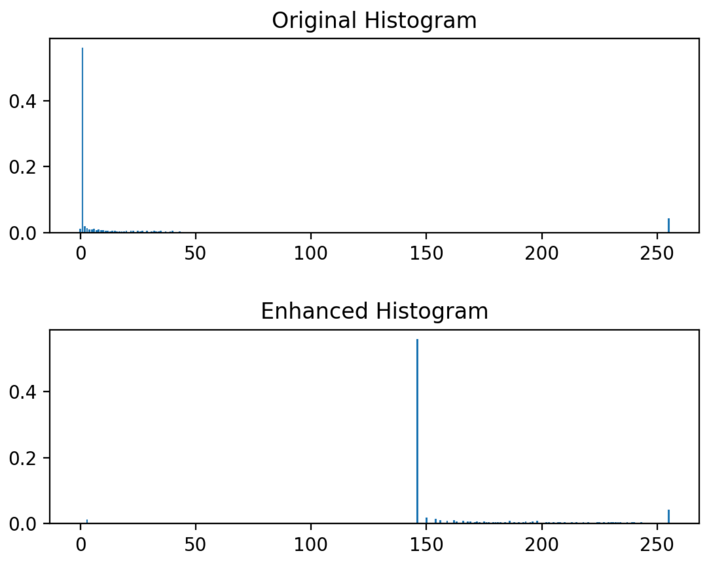
```python
import numpy as np
from matplotlib import pyplot as plt
import math

# given an MxN array of intensity levels, return 1x256 array where each index i indicates
# (the number of pixels with intensity value i)/(total number of pixels)
def makeHisto(img):
    M, N = img.shape
    histo = np.zeros(256)
    # for each intensity value r_k in img[i][j], increase y by 1 at index r_k
    for i in range(0, M):
        for j in range(0, N):
            histo[int(img[i][j])] += 1
    return (histo/(M*N))

# equalizes histogram
def equalizeHisto(histo):
    equalHisto = np.zeros(256)
    equalHisto[0] = y[0]
    for i in range(1, 256):
        equalHisto[i] = equalHisto[i-1] + histo[i]
    return equalHisto

# performs histogram equalization on img and returns resulting image
def equalizeImage(img):
    M, N = img.shape
    product = np.zeros((M, N))
    s = equalizeHisto(makeHisto(img))
    for i in range(0, M):
        for j in range(0, N):
            product[i][j] = math.floor(s[int(img[i][j])] * 255)
    return product

if __name__ == '__main__':
```

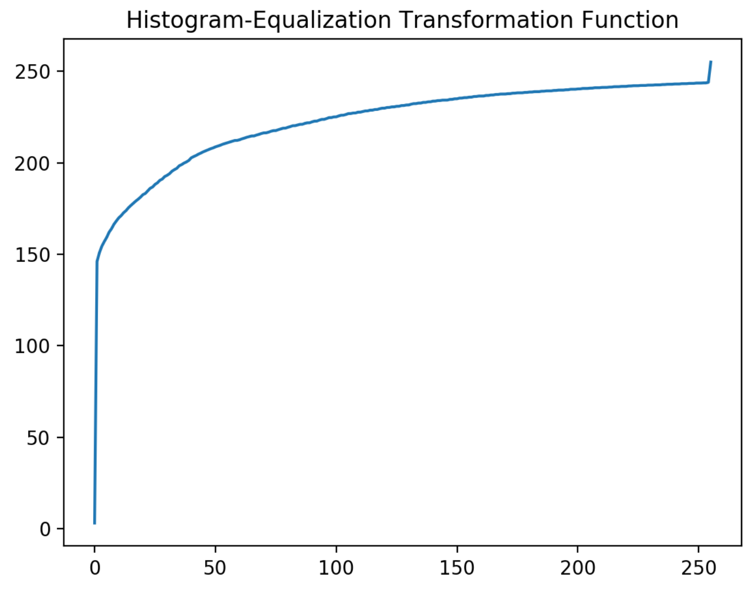
```python
# reads file image into numpy array
img = plt.imread('../images/Fig3.08(a).jpg', 0)
M, N = img.shape

x = list(range(0, 256))  # x-axis
y = makeHisto(img)  # histogram of img
z = equalizeHisto(y) # equalized histogram

enhanced_img = equalizeImage(img)
y2 = makeHisto(enhanced_img) #histogram of enhanced img

## HISTOGRAM ##
fig, axes = plt.subplots(2)
axes[0].set_title('Original Histogram')
axes[0].bar(x, y)
axes[1].set_title('Enhanced Histogram')
axes[1].bar(x, y2)
fig.subplots_adjust(hspace=0.5)

## IMAGES ##
fig2, axes2 = plt.subplots(2)
axes2[0].set_title('Original Image')
axes2[0].imshow(img, cmap='gray')
axes2[1].set_title('Enhanced Image')
axes2[1].imshow(enhanced_img, cmap='gray')
fig2.subplots_adjust(hspace=0.5)

## TRANSFORMATION FUNCTION ##
fig3, axes3 = plt.subplots(1)
axes3.set_title('Histogram-Equalization Transformation Function')
axes3.plot(x, z*255)

plt.show()
```

4c)


Original Image
0
100
200
300
400
0
200


Enhanced Image
0
100
200
300
400
0
200


Original Histogram
0.4
0.2
0.0
0
50
100
150
200
250
Enhanced Histogram
0.4
0.2
0.0
0
50
100
150
200
250


Histogram-Equalization Transformation Function
250
200
150
100
50
0
0
50
100
150
200
250