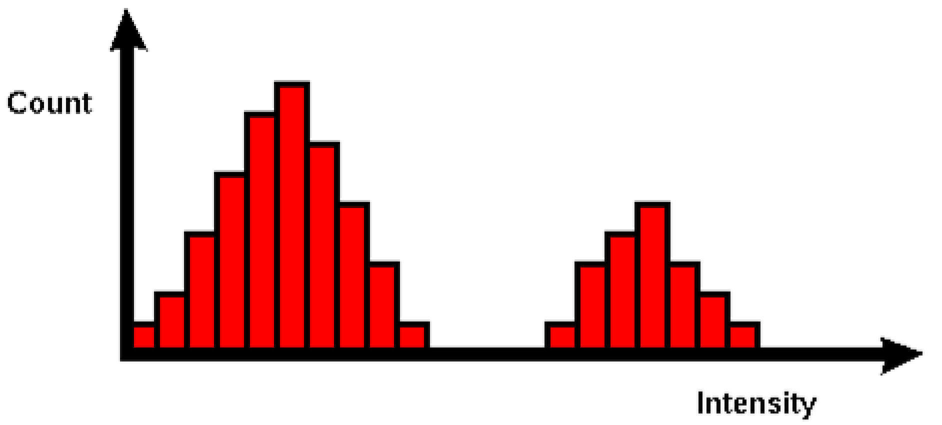
# Image processing concepts

Part 2

# Histogram

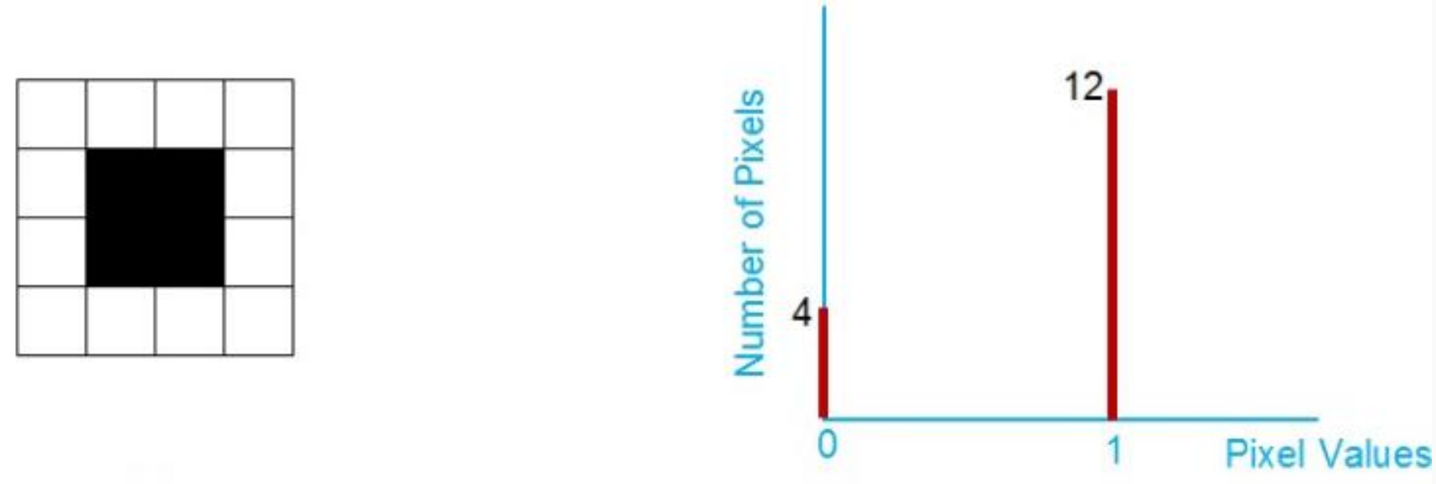
It is a graph showing the number of pixels in an image at each different intensity value found in that image, in other words it shows the **predominant intensities** of an image.

***x-axis*** is the **intensity value** from 0 to 255
***y-axis*** is the **number of pixels** with that intensity value, these values varies depending on the number of the pixels in the image and how their intensities are distributed



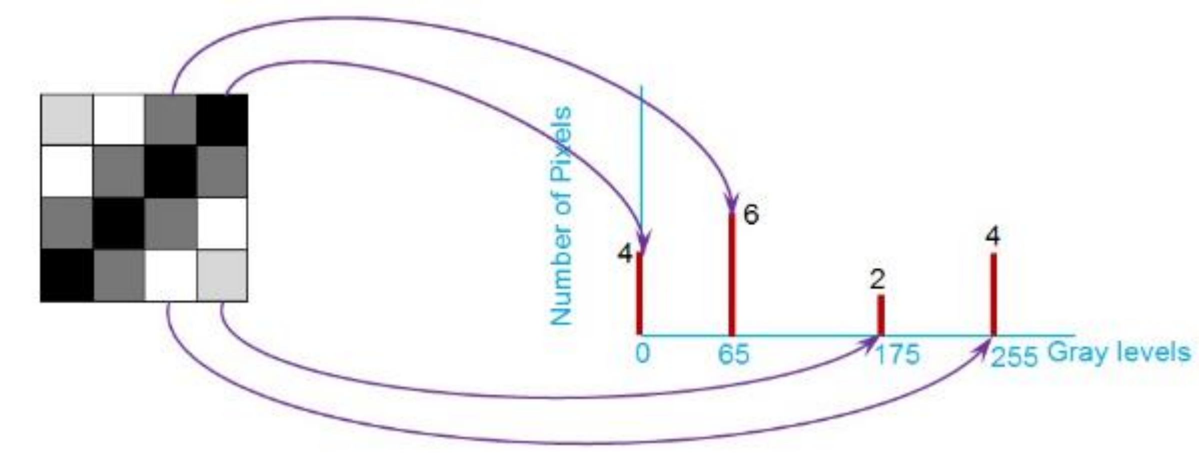
# Example 1

Histogram of simple 4 × 4 **black-and-white** image



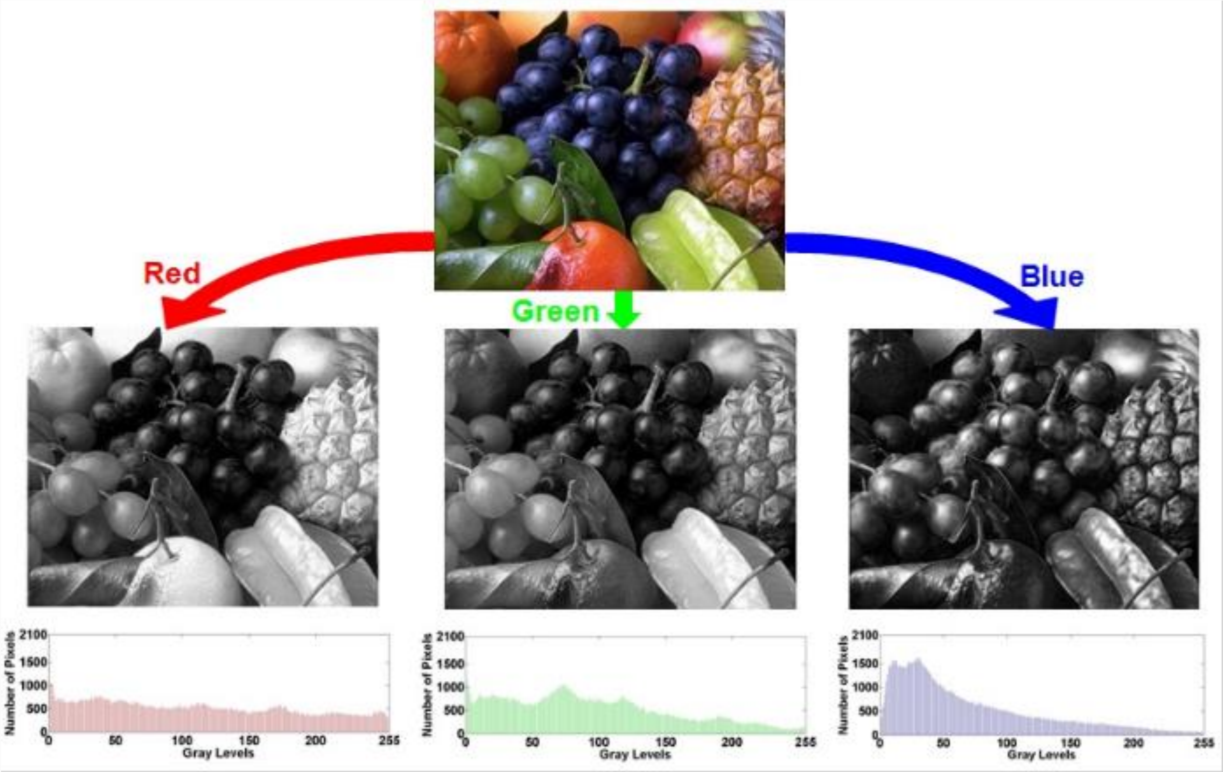
# Example 2

Histogram of **Grayscale** image



# Example 3

Histogram of a **color** image

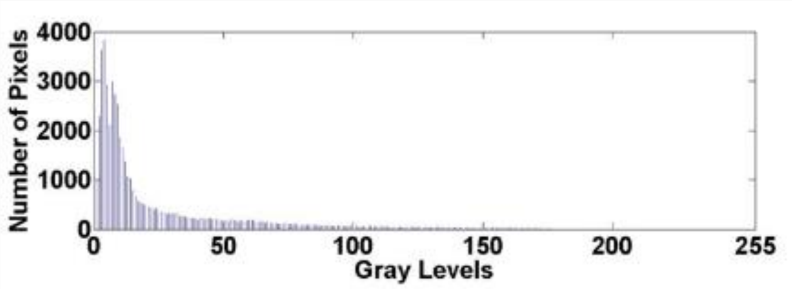


# Histogram Advantages

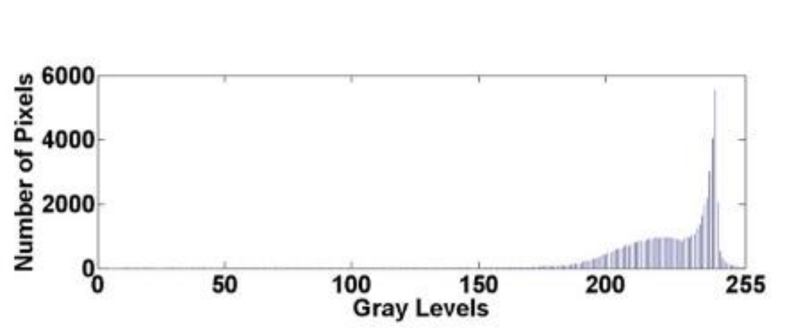
A mere look at the histogram reveals important facts

1. **Image brightness**

Histogram of a **dark** image: values are concentrated toward the **left**



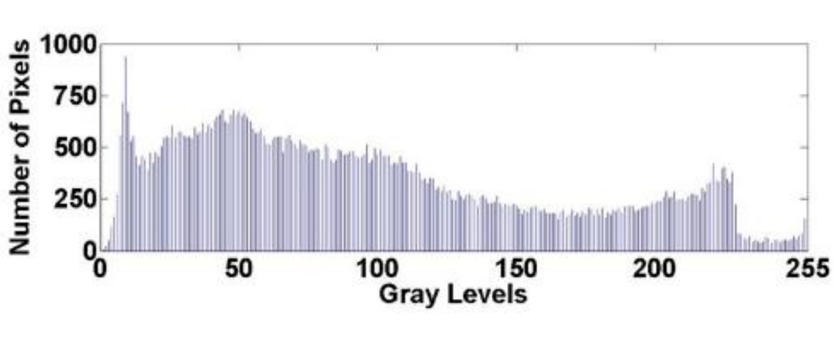
Histogram of a **bright** image: values are concentrated toward the **right**



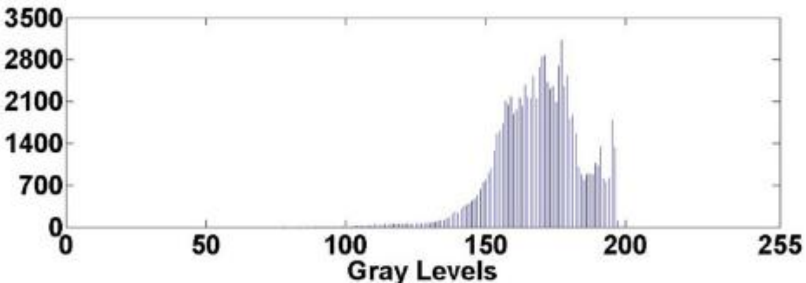
# Histogram Advantages

## 2. Contrast of the image

Histogram of a **high-contrast** image: pixel counts evenly **cover a broad range** of grayscale levels.



Histogram of a **low-contrast** image: Pixel counts that are **restricted** to a smaller range indicate low contrast



# Image Entropy

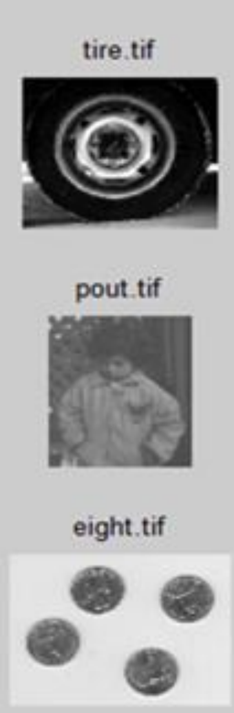
Entropy is a statistical measure of **randomness** that can be used to characterize the input image.
Entropy formula:

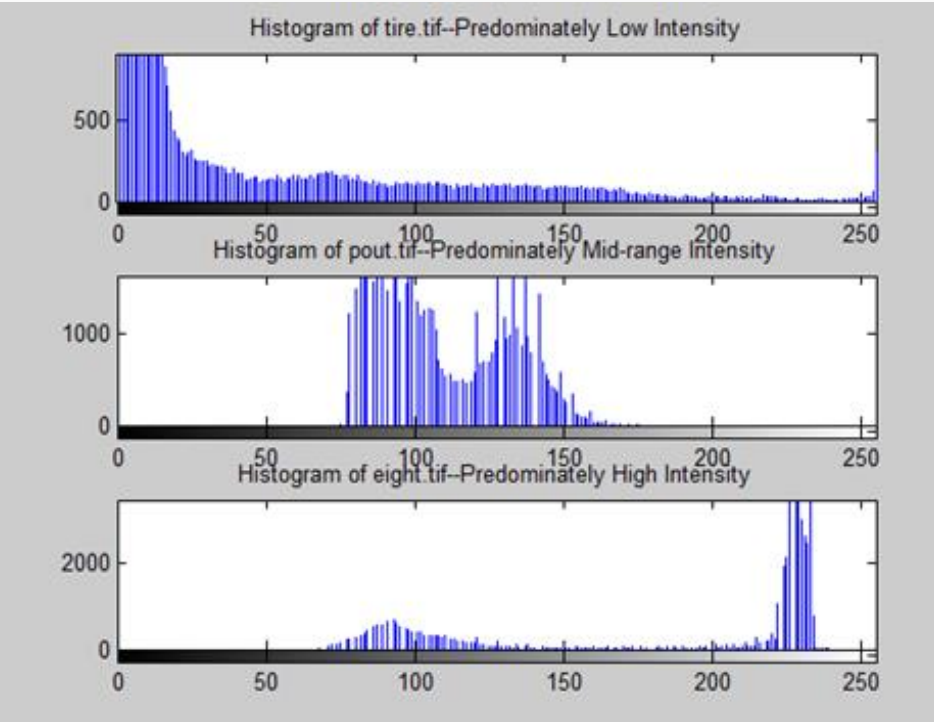
$$e = -\sum_{i=0}^{n-1} p_i \log p_i$$

**n** is the number of gray levels (256 for 8-bit images),
**$p_i$** is the probability of a pixel having gray level intensity

# Image Entropy

Suppose we have these 3 images and their related histograms,
We calculate entropy for each image and get three values : **5.75 , 4.44 , 6.95**
Which value relate to each of these images**??** And Why **??**

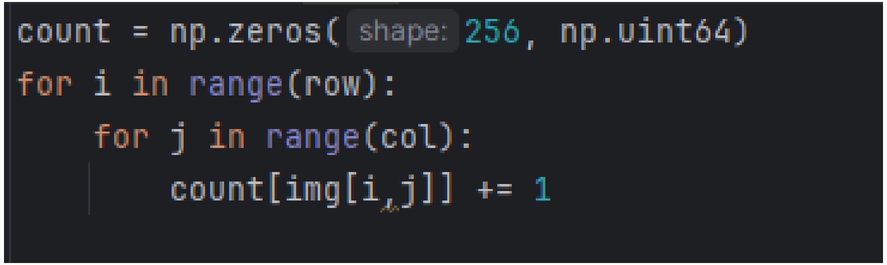




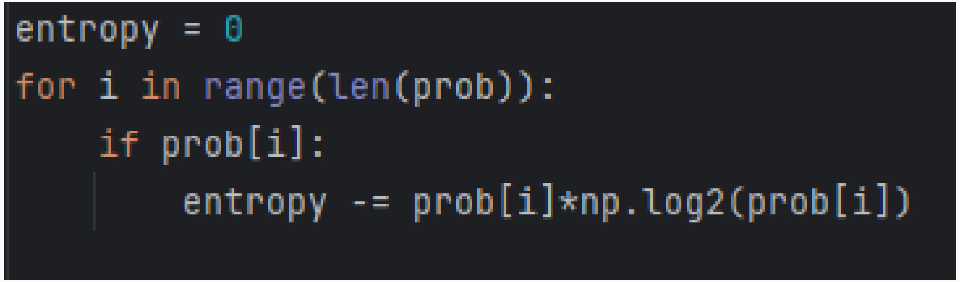
# Image Entropy

Write java code to calculate histogram, draw it and check the previous values.

Here is Snippet of  code



```
count = np.zeros( shape: 256, np.uint64)
for i in range(row):
    for j in range(col):
        count[img[i,j]] += 1
```



```
entropy = 0
for i in range(len(prob)):
    if prob[i]:
        entropy -= prob[i]*np.log2(prob[i])
```