Assignment one

Samih Warwar 324888155

Merry Shalabi 324007202

problem 1:

a) Higher megapixels allow for larger prints or higher resolution displays without sacrificing image quality. That's because more megapixels can capture finer details in an image, but in order to achieve high resolution images we should use higher megapixel camera that have larger sensor size and it may be more expensive to produce, in top of that in order to generate or process high resolution images demands more computational power to handle the large amounts of data associated with it.

b) Older computer hardware or less advanced displays might have limited bit depths for representing colors. A lower bit depth means fewer quantization levels for each color channel, leading to a more noticeable impact on image quality. Common bit depths include 8-bit (256 levels) or even lower in some historical systems. In addition, stronger quantization requires less computational effort for processing, that's why in systems with limited processing power a balance between image quality and processing speed had to be struck.
And when it comes to screens Lower resolution screens may have been more forgiving of stronger quantization since the individual pixels are larger and the imperfections are less noticeable.

Problem 2:

a) The wavelength (λ) is the distance between two consecutive waves and is given by $\lambda = 1/\text{frequency}$. In the context of the image function $I(X, Y) = \sin(\pi kX)$, where X represents the x-axis, the frequency along the x-axis is k/2, leading to the wavelength along the x-axis as $\lambda = 2/k$. The variable A is not relevant in this equation.

b) The Nyquist sampling theorem states that for a signal with frequency f, it must be sampled at a rate of at least 2f to avoid aliasing. In this context, we can relate the wavelength (λ) to the sampling grid by considering the sampling rate.

The sampling interval (A) and the space between columns (also A) are given, and the sampling rate (1/A) is defined as the number of samples per unit distance. To avoid aliasing and fully restore the sine image, the sampling rate should be at least twice the frequency of the sine wave.

For the given sine wave $I(X, Y) = \sin(\pi kX)$, the frequency is related to the wave number (k) as $f = k/2$. Therefore, the condition for avoiding aliasing is:

Sampling rate $\geq$ 2 * Frequency

$1/A \geq 2 * (k/2)$

$1/A \geq k$

So, the k values that will allow us to fully restore the sine image using the given sampling grid are such that $k \leq 1/A$.

For the two values of A provided ({0.25, 2}), the corresponding k values are:

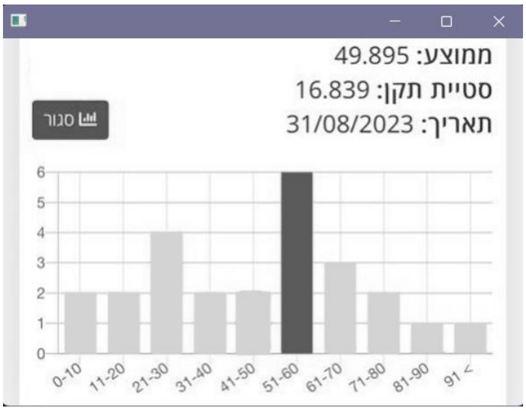
For $A = 0.25$, $k \leq 1/0.25 = 4$

For $A = 2$, $k \leq 1/2 = 0.5$

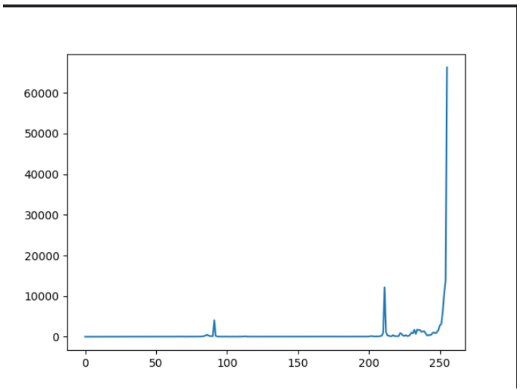
Therefore, the k values for full restoration are $k \leq 4$ for $A = 0.25$ and $k \leq 0.5$ for $A = 2$.

Problem 3:

a) displaying the first image in gray



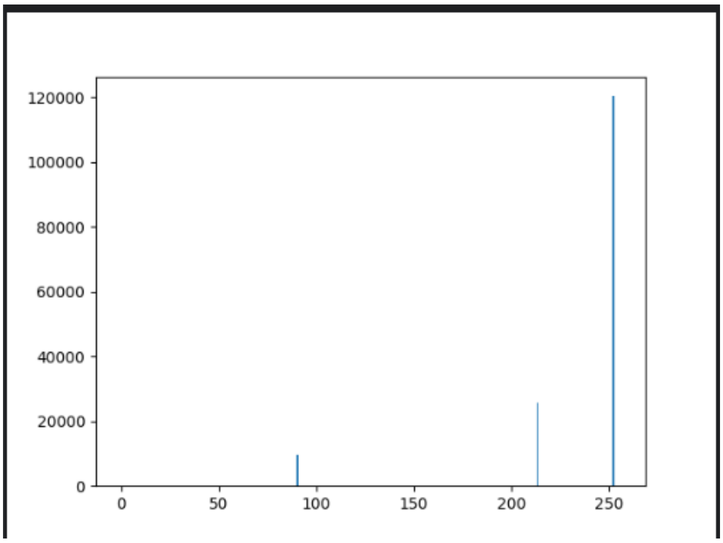
b) here we displayed some histograms



c) Here we took the windows size same as the numbers size and then, to calculate the EMD where each time we calculated the cumulative histogram of the number and the cumulative histogram of the window and calculated the difference between them and checked if its -260<EMD<260.

d) we found that the highest number in the given images located exactly higher than 115 pixel from top and within 0-70 pixels from the right and that's where we run the sliding window starting from 9 till 0 we checked each image

e) we basically tried to choose number of the colors from bottom to up firstly when choosing 1 it will not help because we will have one color picture and when we tried 2 background merged with most of the bars leaving just one bar visible and when choosing 3 we got this histogram



so we chose threshold equal to 240 it worked fine.

f) we can see in the first image that we have in bar number 5 the highest bar and using the get_bar_height we got that its located at pixel 156.

g) here we saved each bar height and calculated the students number in each image and saved them in matrix and then we printed and got as a result:

```
Histogram a.jpg gave 2,2,4,2,2,6,3,2,1,1
Histogram b.jpg gave 6,2,1,1,3,3,6,2,2,2
Histogram c.jpg gave 0,0,0,0,0,0,1,1,1,1
Histogram d.jpg gave 1,0,2,3,4,3,5,5,6,6
Histogram e.jpg gave 2,1,1,3,2,5,1,1,2,2
Histogram f.jpg gave 1,0,1,1,1,4,1,1,2,2
Histogram g.jpg gave 1,1,1,3,1,2,9,3,3,3
```