

ex1

September 4, 2024

1 Ejercicio 1

```
[ ]: from PIL import Image
     from pathlib import Path
     import matplotlib.pyplot as plt
```

```
[ ]: ASSETS_FOLDER_PATH = "../assets"
     OUTPUT_FOLDER_PATH = "../"
```

```
[ ]: Path(OUTPUT_FOLDER_PATH).mkdir(parents=True, exist_ok=True)
```

```
[ ]: rose_b_w_1024x1024 = Image.open(f"{ASSETS_FOLDER_PATH}/Fig0219(rose1024).tif")
     rose_b_w_1024x1024.save(f"{OUTPUT_FOLDER_PATH}/rose_b_w_1024x1024.bmp")
     display(f"{rose_b_w_1024x1024.width}x{rose_b_w_1024x1024.height}")
     display(rose_b_w_1024x1024)
```

'1024x1024'



```
[ ]: building = Image.open(f"{ASSETS_FOLDER_PATH}/building.jpg")  
display(f"{building.width}x{building.height}")  
display(building)
```

'512x384'



```
[ ]: skull_256k = Image.open(f"{ASSETS_FOLDER_PATH}/Fig0221(a)(ctskull-256).tif")  
      display(f"{skull_256k.width}x{skull_256k.height}")  
      display(skull_256k)
```

'374x452'



```
[ ]: def subsampling(image: Image.Image) -> Image.Image:
    result_y = Image.new(mode=image.mode, size=(int(image.width), int(image.
    ↪height/2)))
    result_y_pixels = result_y.load()
    image_pixels = image.load()

    y_res = 0
    for y in range(image.height):
        if y % 2 == 0:
            for x in range(image.width):
```

```

        result_y_pixels[x, y_res] = image_pixels[x, y]
        y_res += 1

    result = Image.new(mode=image.mode, size=(int(image.width/2), int(image.
↪height/2)))
    result_pixels = result.load()
    x_res = 0
    for x in range(result_y.width):
        if x % 2 == 0:
            for y in range(result_y.height):
                result_pixels[x_res, y] = result_y_pixels[x, y]
                x_res += 1

    return result

```

```

[ ]: def replicating(image: Image.Image) -> Image.Image:
    result_y = Image.new(mode=image.mode, size=(int(image.width), int(image.
↪height*2)))
    result_y_pixels = result_y.load()
    image_pixels = image.load()

    y_res = 0
    for y in range(image.height):
        for _ in range(2):
            for x in range(image.width):
                result_y_pixels[x, y_res] = image_pixels[x, y]
                y_res += 1

    result = Image.new(mode=image.mode, size=(int(image.width*2), int(image.
↪height*2)))
    result_pixels = result.load()
    x_res = 0
    for x in range(result_y.width):
        for _ in range(2):
            for y in range(result_y.height):
                result_pixels[x_res, y] = result_y_pixels[x, y]
                x_res += 1

    return result

```

```

[ ]: def scale_down_luminance(image: Image.Image, levels: int) -> Image.Image:
    result = image.copy()
    result_pixels = result.load()
    for x in range(result.width):
        for y in range(result.height):
            result_pixels[x, y] = round(round(result_pixels[x, y]*(levels-1) /_
↪255)*(255/(levels-1)))

```

```
return result
```

```
[ ]: rose_b_w_512x512 = subsampling(rose_b_w_1024x1024)
rose_b_w_512x512.save(f"{OUTPUT_FOLDER_PATH}/rose_b_w_512x512.bmp")
display(f"{rose_b_w_512x512.width}x{rose_b_w_512x512.height}")
display(rose_b_w_512x512)
```

'512x512'



```
[ ]: rose_b_w_256x256 = subsampling(rose_b_w_512x512)
rose_b_w_256x256.save(f"{OUTPUT_FOLDER_PATH}/rose_b_w_256x256.bmp")
display(f"{rose_b_w_256x256.width}x{rose_b_w_256x256.height}")
display(rose_b_w_256x256)
```

'256x256'



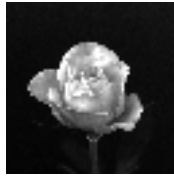
```
[ ]: rose_b_w_128x128 = subsampling(rose_b_w_256x256)
rose_b_w_128x128.save(f"{OUTPUT_FOLDER_PATH}/rose_b_w_128x128.bmp")
display(f"{rose_b_w_128x128.width}x{rose_b_w_128x128.height}")
display(rose_b_w_128x128)
```

'128x128'



```
[ ]: rose_b_w_64x64 = subsampling(rose_b_w_128x128)
rose_b_w_64x64.save(f"{OUTPUT_FOLDER_PATH}/rose_b_w_64x64.bmp")
display(f"{rose_b_w_64x64.width}x{rose_b_w_64x64.height}")
display(rose_b_w_64x64)
```

'64x64'



```
[ ]: rose_b_w_32x32 = subsampling(rose_b_w_64x64)
rose_b_w_32x32.save(f"{OUTPUT_FOLDER_PATH}/rose_b_w_32x32.bmp")
display(f"{rose_b_w_32x32.width}x{rose_b_w_32x32.height}")
display(rose_b_w_32x32)
```

'32x32'



```
[ ]: rose_b_w_from_512x512 = replicating(rose_b_w_512x512)
rose_b_w_from_512x512.save(f"{OUTPUT_FOLDER_PATH}/rose_b_w_from_512x512.bmp")
display(f"{rose_b_w_from_512x512.width}x{rose_b_w_from_512x512.height}")
display(rose_b_w_from_512x512)
```

'1024x1024'



```
[ ]: rose_b_w_from_256x256 = replicating(replicating(rose_b_w_256x256))
rose_b_w_from_256x256.save(f"{OUTPUT_FOLDER_PATH}/rose_b_w_from_256x256.bmp")
display(f"{rose_b_w_from_256x256.width}x{rose_b_w_from_256x256.height}")
display(rose_b_w_from_256x256)
```

'1024x1024'



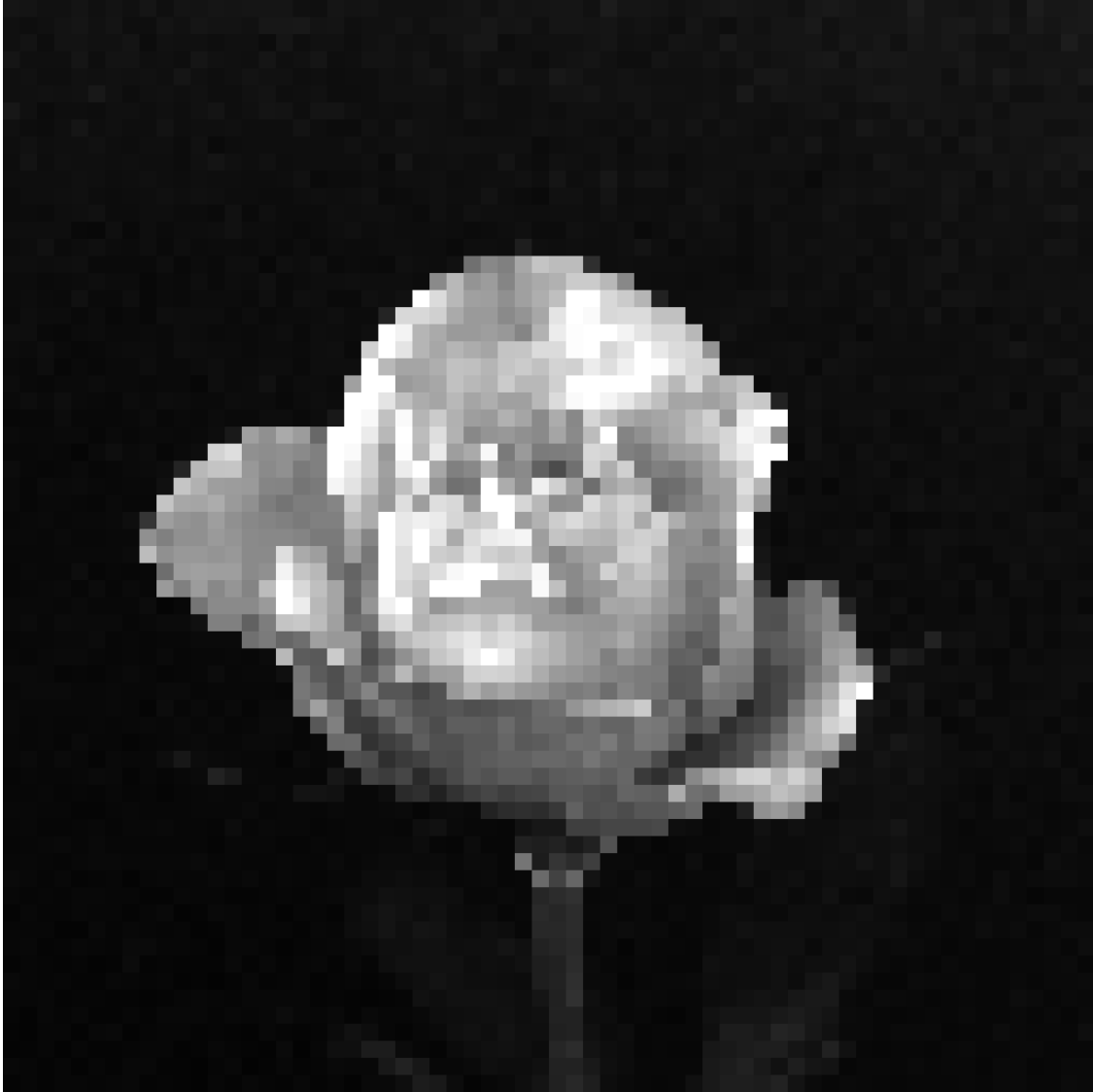
```
[ ]: rose_b_w_from_128x128 = replicating(replicating(replicating(rose_b_w_128x128)))  
rose_b_w_from_128x128.save(f"{OUTPUT_FOLDER_PATH}/rose_b_w_from_128x128.bmp")  
display(f"{rose_b_w_from_128x128.width}x{rose_b_w_from_128x128.height}")  
display(rose_b_w_from_128x128)
```

'1024x1024'



```
[ ]: rose_b_w_from_64x64 =  
    ↪replicating(replicating(replicating(replicating(rose_b_w_64x64))))  
rose_b_w_from_64x64.save(f"{OUTPUT_FOLDER_PATH}/rose_b_w_from_64x64.bmp")  
display(f"{rose_b_w_from_64x64.width}x{rose_b_w_from_64x64.height}")  
display(rose_b_w_from_64x64)
```

'1024x1024'



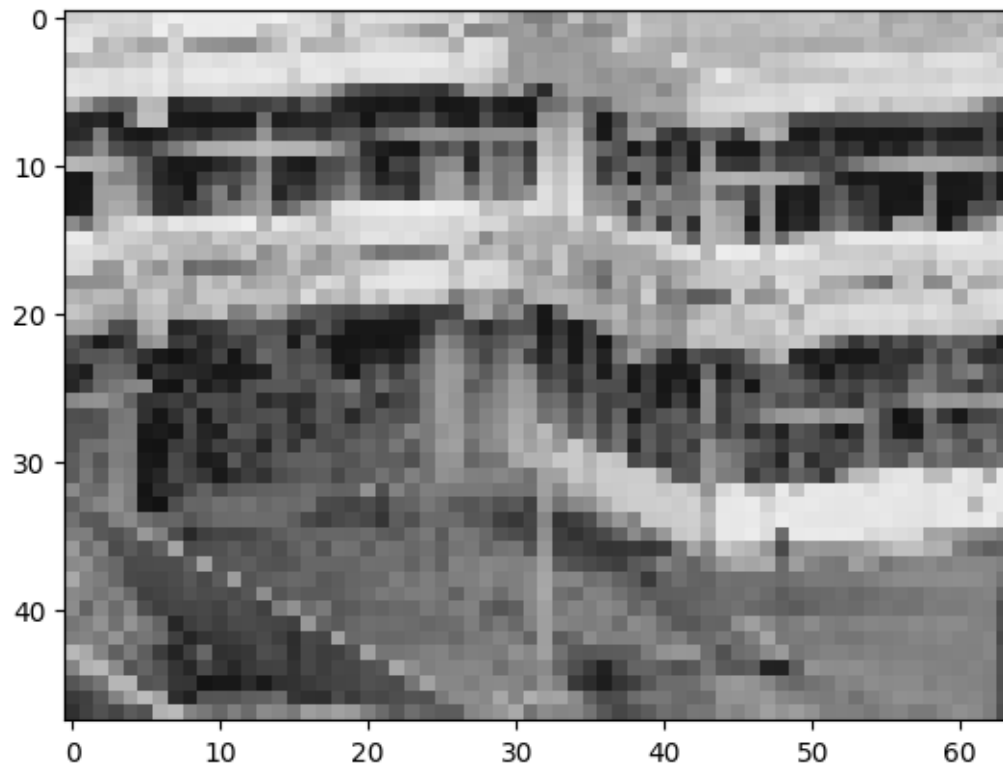
```
[ ]: rose_b_w_from_32x32 =  
    ↪replicating(replicating(replicating(replicating(replicating(rose_b_w_32x32))))  
rose_b_w_from_32x32.save(f"{OUTPUT_FOLDER_PATH}/rose_b_w_from_32x32.bmp")  
display(f"{rose_b_w_from_32x32.width}x{rose_b_w_from_32x32.height}")  
display(rose_b_w_from_32x32)
```

'1024x1024'



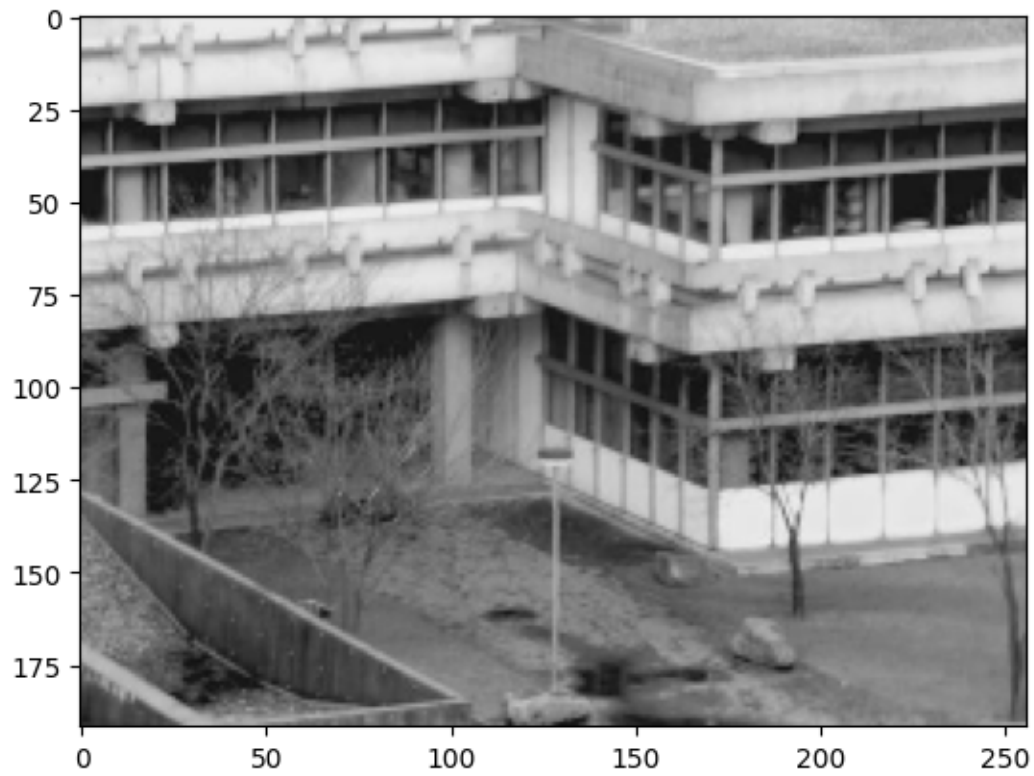
```
[ ]: building_48x64 = subsampling(subsampling(subsampling(building)))  
building_48x64.save(f"{OUTPUT_FOLDER_PATH}/building_48x64.bmp")  
plt.imshow(building_48x64, cmap="gray", vmin=0, vmax=255)
```

```
[ ]: <matplotlib.image.AxesImage at 0x79b6c7f18ee0>
```



```
[ ]: building_192x256 = subsampling(building)
      building_192x256.save(f"{OUTPUT_FOLDER_PATH}/building_192x256.bmp")
      plt.imshow(building_192x256, cmap="gray", vmin=0, vmax=255)
```

```
[ ]: <matplotlib.image.AxesImage at 0x79b6c7fe6260>
```



```
[ ]: skull_128k = scale_down_luminance(skull_256k, 128)
skull_128k.save(f"{OUTPUT_FOLDER_PATH}/skull_128k.bmp")
display(skull_128k)
```



```
[ ]: skull_64k = scale_down_luminance(skull_256k, 64)
skull_64k.save(f"{OUTPUT_FOLDER_PATH}/skull_64k.bmp")
display(skull_64k)
```




```
[ ]: skull_32k = scale_down_luminance(skull_256k, 32)
skull_32k.save(f"{OUTPUT_FOLDER_PATH}/skull_32k.bmp")
display(skull_32k)
```



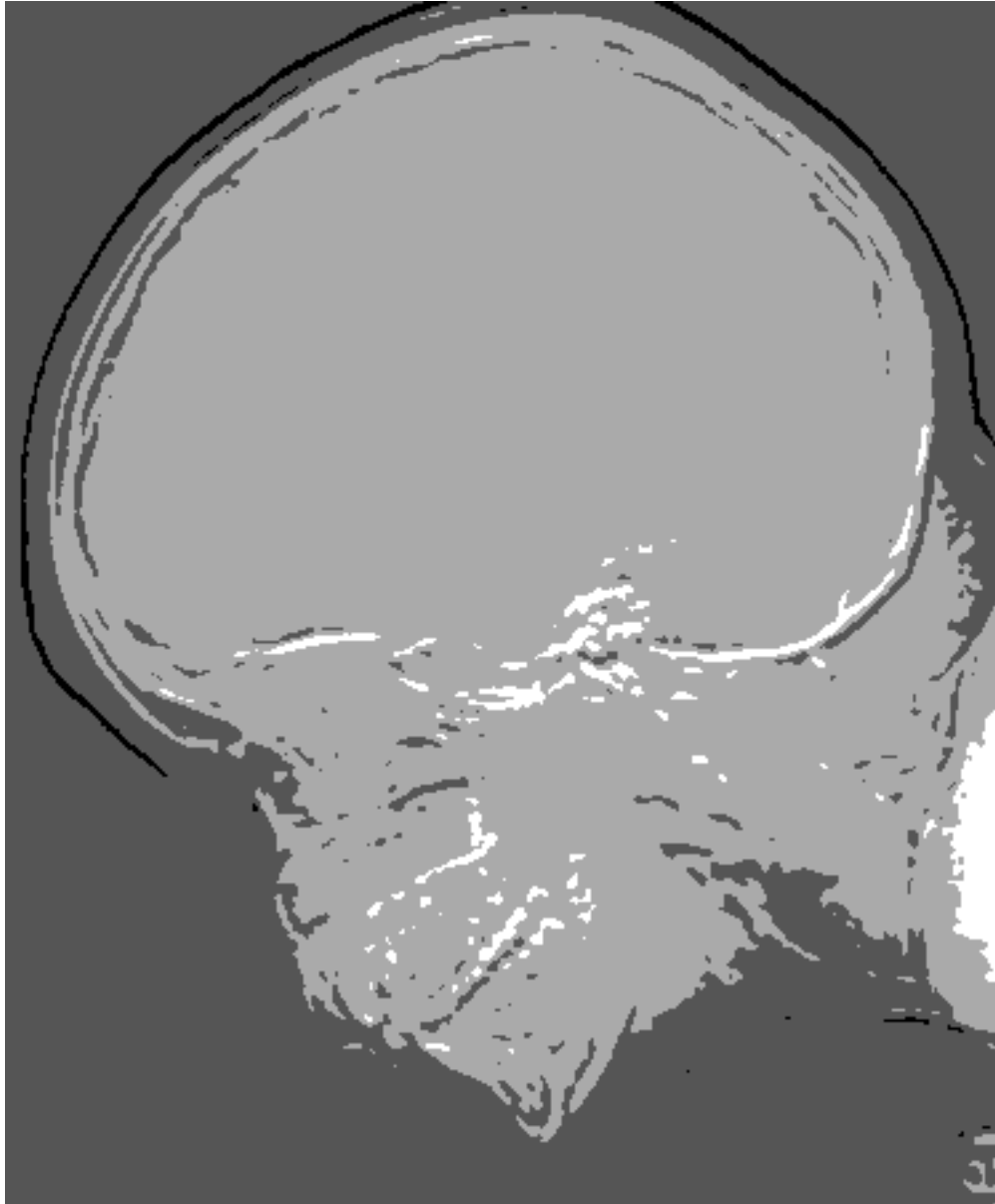
```
[ ]: skull_16k = scale_down_luminance(skull_256k, 16)
skull_16k.save(f"{OUTPUT_FOLDER_PATH}/skull_16k.bmp")
display(skull_16k)
```



```
[ ]: skull_8k = scale_down_luminance(skull_256k, 8)
skull_8k.save(f"{OUTPUT_FOLDER_PATH}/skull_8k.bmp")
display(skull_8k)
```



```
[ ]: skull_4k = scale_down_luminance(skull_256k, 4)
skull_4k.save(f"{OUTPUT_FOLDER_PATH}/skull_4k.bmp")
display(skull_4k)
```



```
[ ]: skull_2k = scale_down_luminance(skull_256k, 2)
skull_2k.save(f"{OUTPUT_FOLDER_PATH}/skull_2k.bmp")
display(skull_2k)
```

