

Practice 1: JPEG

1) Start a script called `rgb_yuv.py` and create a translator from 3 values in RGB into the 3 YUV values, plus the opposite operation.

You can choose the 3 values, or open them from a text file, receive it from the command line... feel free.

This exercise is attached in the practice folder.

2) Use ffmpeg to resize images into lower quality.

Do screenshots of your operations and upload the results.

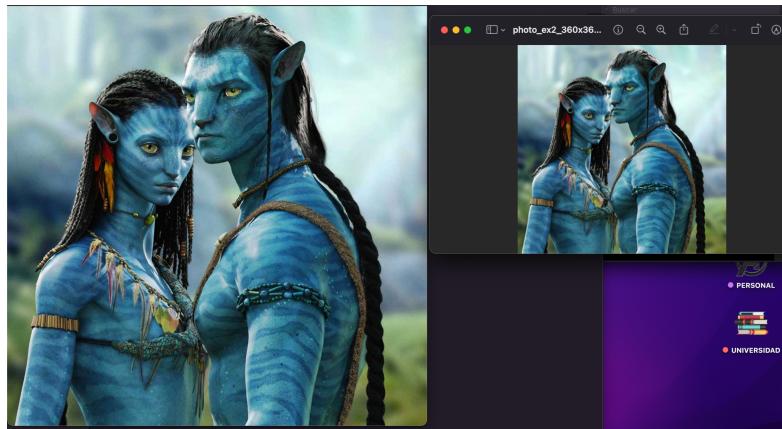
The command that I have used to convert the image of 1600x1600 to an image of 360x360 is:

```
ffmpeg -i photo_ex2.jpeg -vf scale=360:360 photo_ex2_360x360.png
```

This is the first screenshot of the result.



The dimensions of the image have changed, therefore, we have fewer pixels. If we try to make the character the same size we see that we lose quality, since the pixel density is lower.



Here we have a comparison between the images of different sizes.

3) Use FFMPEG to transform the Lenna image into b/w. Do the hardest compression you can and comment on the results.

```
ffmpeg -i photo_ex2.jpeg photo_compressed_ex3.jpeg; ffmpeg -i  
photo_compressed_ex3.jpeg -vf hue=s=0 photo_compressed_ex3_bw.jpeg
```

For the following exercise, what I have done has been to first compress the photo and obtain a smaller photo. The first one (the original) occupies 306,275 bytes and the compressed one 147,338 bytes.

In the screenshots attached below you can see that loss of information. If I zoom in the image, in the second one we can see “squares” and in the first one these are not so noticeable.



Finally, with the following line the compressed image has been converted to black and white. This image occupies even less.



4) Create a script which contains a function which applies a run-length encoding from a series of bytes given.

This exercise is attached in the practice folder.

5) Create a script which can convert, can decode (or both) an input using the DCT. Not necessary a JPG encoder or decoder. A script only about DCT is OK too

This exercise is attached in the practice folder.