S1 – JPEG, JPEG2000 and FFMpeg

- Use PYTHON only (there might be exceptions)

- Be creative! Feel free to type code as you want

- Don’t forget to comment your code to make it understandable if needed

- PEP8 it’s a plus https://www.python.org/dev/peps/pep-0008/

- **It's recommended to work with PyCharm or any other IDE**

- **You can INTERPRET as you want the following exercises**

- **It's ALLOWED to COPY from the internet if the script works. Not allowed to copy from mates AND DO NOT USE CHATGPT, COPILOT or any AI**

- Deliver everything inside a Github link! then share this link in Aula Global

- Work with your bestie together. Better github history will increase marks

FFmpeg

(From the Wiki)

FFmpeg is a free and open-source project consisting of a vast software suite of libraries and programs for handling video, audio, and other multimedia files and streams. At its core is the FFmpeg program itself, designed for command-line-based processing of video and audio files, and widely used for format transcoding, basic editing (trimming and concatenation), video scaling, video post-production effects, and standards compliance (SMPTE, ITU).

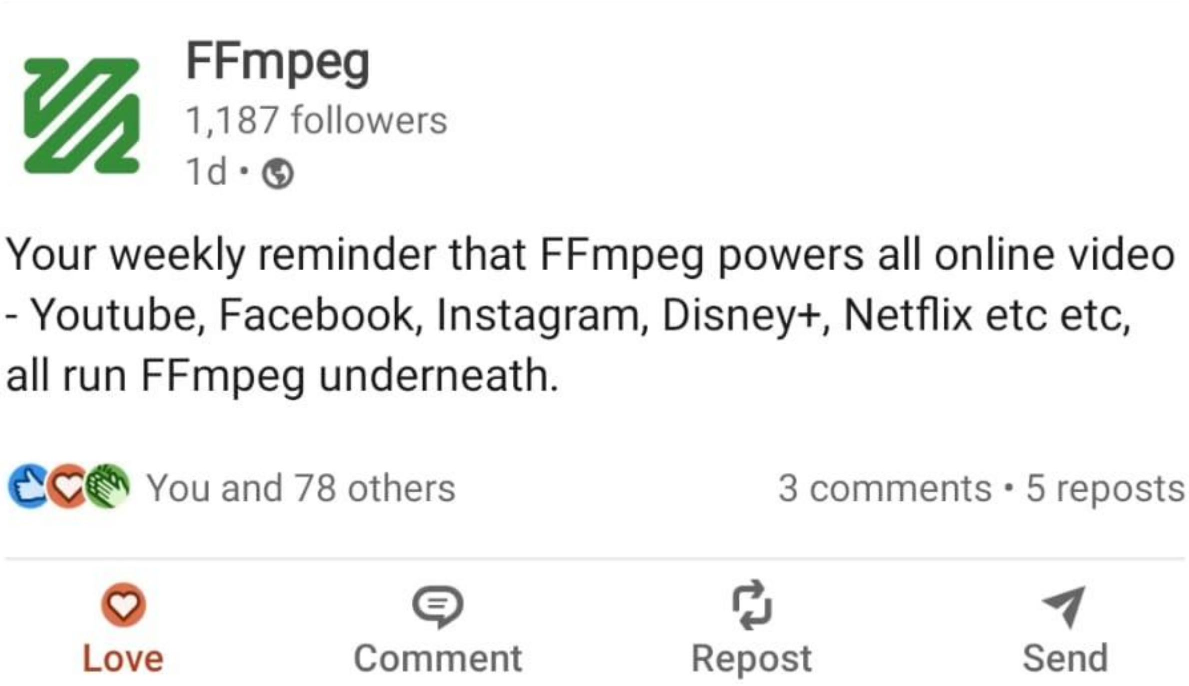
FFmpeg includes libavcodec, an audio/video codec library used by many commercial and free software products, libavformat (Lavf),[6] an audio/video container mux and demux library, and the core ffmpeg command line program for transcoding multimedia files.

- It began as a small software to test the libavcodec library in Linux

- It's the MOST USED software inside the tech video industry

- It can be shitty sometimes, but as it works with EVERYTHING everybody uses them

- It's open source and mantained by the community



FFmpeg

1,187 followers

1d •

Your weekly reminder that FFmpeg powers all online video - Youtube, Facebook, Instagram, Disney+, Netflix etc etc, all run FFmpeg underneath.

You and 78 others

3 comments • 5 reposts

1.-)

- Compile & install the lastest version on your laptop/desktop/whatever with the command line
- Important! ensure you are installing ALL the recent libraries
- Run ‘`ffmpeg`’ and upload a screenshot of the first line
- If you don’t finish, type ‘`history`’ and upload the snapshots to prove you’ve tried to

2) Start a script called *first_seminar.py* . Then create a class and a method , which is 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 command line... feel free.

3) Use ffmpeg to resize images into lower quality. Use any image you like

Now, create a method in previous script to automatise this order.

4) Create a method called *serpentine* which should be able to read the bytes of a JPEG file in the serpentine way we saw.

5) Use FFMPEG to transform the previous image into b/w. Do the hardest compression you can.

Add everything into a new method and comment the results

5) Create a method which applies a run-lenght encoding from a series of bytes given.

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

7) Create a class which can convert, can decode (or both) an input using the DWT. Not necessary a JPEG2000 encoder or decoder. A class only about DWT is OK too

8) Use any AI (YES, you can NOW, you lazy!) to create UNIT TESTS to your code, for each method and class

If the code is too much poor, try to improve it a bit

Thanks

franciscojavier.brines@upf.edu