

P2 Video Encoding Systems.

Sebastian R. Ovelar Anderson

NIA:206384

In this Lab we will work with python and video files.

- 1) Create a python script able to parse the 'ffmpeg -i BBB.mp4' file, which can mark at least 3 relevant data from the container

I interpreted this exercise in a way that we have to parse the BBB.mp4 file and then save in python at least 3 relevant data like the resolution, codecs, etc...

This exercise is done in the E1.py. I mainly used the ffprobe commands to extract this and the subprocess library to read it from the terminal. The result was:

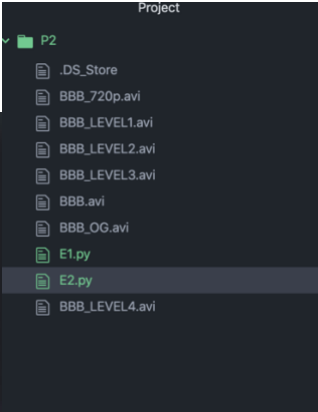
```
P2 - zsh - 80x24
Last login: Sat Nov 28 02:04:57 on ttys000
(base) sebastianderson@MacBook-Pro-de-Sebastian ~ % cd ~/Desktop/Uni/ESAV/P2/
(base) sebastianderson@MacBook-Pro-de-Sebastian P2 % python E1.py
Video Code: b'ac3\n'
Audio Code: b'ac3\n'
Width x Height: b'1920x1080\n'
(base) sebastianderson@MacBook-Pro-de-Sebastian P2 %
```

If you tried to run the file it may not work because you need the BBB.avi file in the same folder.

- 2) Create a python script able to rename the 5 quality outputs of the BBB that you did in last seminar

This is done in E2.py file. With the os.rename() command. As we see the file have changed name.

```
P2 - zsh - 80x24
Last login: Fri Nov 27 23:18:09 on ttys000
(base) sebastianderson@MacBook-Pro-de-Sebastian ~ % python E2.py
python: can't open file 'E2.py': [Errno 2] No such file or directory
(base) sebastianderson@MacBook-Pro-de-Sebastian ~ % cd ~/Desktop/Uni/ESAV/P2/
(base) sebastianderson@MacBook-Pro-de-Sebastian P2 % python E2.py
File "E2.py", line 1
import numpy as np
^
SyntaxError: invalid syntax
(base) sebastianderson@MacBook-Pro-de-Sebastian P2 % python E2.py
Traceback (most recent call last):
  File "E2.py", line 5, in <module>
    os.rename("BBB_160x102.avi", "BBB_LEVEL1.avi")
FileNotFoundError: [Errno 2] No such file or directory: 'BBB_160x102.avi' -> 'BBB_LEVEL1.avi'
(base) sebastianderson@MacBook-Pro-de-Sebastian P2 % python E2.py
(base) sebastianderson@MacBook-Pro-de-Sebastian P2 %
```



If you tried to run the file it may not work because you need all the BBB files in the same folder.

3) Create a python script able to resize (resolution change) of any input given

I created a function able to resize any file in E3.py. The function takes as argument the filename, the size in which you want to resize it and the output filename. As we see this is done when we tried the function.

```
resize(BBB_OG.avi,160,prueba.avi)
NameError: name 'BBB_OG' is not defined
(base) sebastianderson@MacBook-Pro-de-Sebastian P2 % python E3.py
ffmpeg version N-99847-gcfddec Copyright (c) 2000-2020 the FFmpeg developers
  built with Apple clang version 12.0.0 (clang-1200.0.32.21)
  configuration: --prefix=/Users/sebastianderson/ffmpeg_build --pkg-config-flags=--static --extra-cflags=-I/Users/sebastianderson/ffmpeg_build/include --extra-ldflags=-L/Users/sebastianderson/ffmpeg_build/lib --extra-libs=-lpthread -lm --bindir=/Users/sebastianderson/bin --enable-gpl --enable-gnutls --enable-libaom --enable-libbass --enable-libfdk-aac --enable-libfreetype --enable-libmp3lame --enable-libopus --enable-libsvtav1 --enable-libvorbis --enable-libvpx --enable-libx264 --enable-libx265 --enable-nonfree
  libavutil      56. 60.100 / 56. 60.100
  libavcodec     58.112.101 / 58.112.101
  libavformat    58. 64.100 / 58. 64.100
  libavdevice    58. 11.102 / 58. 11.102
  libavfilter    7. 88.102 / 7. 88.102
  libswscale     5.  8.100 / 5.  8.100
  libswresample  3.  8.100 / 3.  8.100
  libpostproc   55.  8.100 / 55.  8.100
Input #0, avi, from 'BBB_OG.avi':
  Metadata:
    encoder      : Lavf58.64.100
  Duration: 00:00:13.96, start: 0.000000, bitrate: 22382 kb/s
    Stream #0:0 Video: mpeg4 (Simple Profile) (FMP4 / 0x34504D46), yuv420p, 1920x1080 [SAR 1:1 DAR 16:9], 21985 kb/s, 24 fps, 24 tbr, 24 tbn, 24 tbc
    Stream #0:1 Audio: ac3 ([0] [0] [0] / 0x2000), 48000 Hz, 5.1(side), fltp, 448 kb/s
  Stream mapping:
    Stream #0:0 -> #0:0 (mpeg4 (native) -> mpeg4 (native))
    Stream #0:1 -> #0:1 (ac3 (native) -> mp3 (libmp3lame))
Press [q] to stop, [?] for help
[mpeg4 @ 0x7f9c3000ea00] too many threads/slices (9), reducing to 8
Output #0, avi, to 'prueba.avi':
  Metadata:
    ISFT        : Lavf58.64.100
  Stream #0:0 Video: mpeg4 (FMP4 / 0x34504D46), yuv420p(tv, progressive), 160x120 [SAR 1:1 DAR 4:3], q=2-31, 200 kb/s, 24 fps, 24 tbn, 24 tbc
  Metadata:
    encoder      : Lavc58.112.101 mpeg4
  Side data:
    cpb: bitrate max/min/avg: 0/0/200000 buffer size: 0 vbv_delay: N/A
    Stream #0:1 Audio: mp3 (libmp3lame) (U[0] [0] [0] / 0x0055), 48000 Hz, stereo, fltp
  Metadata:
    encoder      : Lavc58.112.101 libmp3lame
frame= 101 fps=0.0 q=3.1 size=      256kB time=00:00:07.80 bitrate= 268.9kbits/sframe= 333 fps=0.0 q=2.9 Lsize=      727kB time=00:00:13.95 bitrate= 426.8kbits/s speed=16.2x
video:478kB audio:218kB subtitle:0kB other streams:0kB global headers:0kB muxing overhead: 4.514442%
(base) sebastianderson@MacBook-Pro-de-Sebastian P2 %
```

4) Create a python script able to transcode the input into an output with another codec that we've seen in the Theory class.

I created a function in E4.py called transcodeh264(). This function allows you to change the codec to h264. As input, you need to add the filename and the output filename. As we can see in the 2 images the first is when we encode in h264 and the second show us that the output is indeed now in h264.

```

(base) sebastianderson@MacBook-Pro-de-Sebastian P2 % ffmpeg -i prueba2.mp4
ffmpeg version N-99847-gcfdddec Copyright (c) 2000-2028 the FFmpeg developers
  built with Apple clang version 12.0.0 (clang-1200.0.32.21)
  configuration: --prefix=/Users/sebastianderson/ffmpeg_build --pkg-config-flags=--static --extra-flags=/Users/sebastianderson/ffmpeg_build/include --extra-ldflags=-lusers/sebastianderson/ffmpeg_build/lib --extra-libs=-lstdc++-lib -lstdc++-lib -enable-libaom -enable-libdav1d -enable-libfontconfig -enable-libfreetype -enable-libgsm -enable-liblame -enable-liblibvorbis -enable-libopus -enable-librtmp -enable-libssh -enable-libtheora -enable-libvpx -enable-libx264 -enable-libx265 -enable-nonfree
  libavcodec 58.132.101 / 58.112.101
  libavformat 58.140 / 58. 64.100
  libavdevice 58.110 / 58. 11.102
  libavfilter 7.80.102 / 7. 80.102
  libswscale 5. 8.100 / 5. 8.100
  libswresample 3. 8.100 / 3. 8.100
  libpostproc 55. 8.100 / 55. 8.100
  requested Channel Layout for Input Stream #0.1: 6.1
Input #0, mov,mp4,ma3,3gp,3g2,m2, from 'prueba2.mp4':
Metadata:
  major_brand      : isom
  minor_version    : 012
  compatible_brands: isomisoavc1mp41
  encoder         : LavF58.64.100
Duration: 00:00:13.96, start: 0.000000, bitrate: 6977 kb/s
Stream #0(u): Video: h264 (high) (avc1 / 0x33376d3e), yuv420p, 1920x1080 (SAR 1:1 DAR 16:9), 6078 kb/s, 24 fps, 24 tbr, 12288 tbn, 48 tbc (default)
Metadata:
  handler_name     : VideoHandler
Stream #0(l): Audio: aac (LC) (mp4a / 0x6d34786d), 48000 Hz, S, 1, fltp, 394 kb/s (default)
Metadata:
  handler_name     : SoundHandler
At least one output file must be specified
(base) sebastianderson@MacBook-Pro-de-Sebastian P2 %

```

For this I created an interactive file where you can ask to the terminal to do something either change the size or the codec. And then you can also choose the new resolution of the video.

```
(base) sebastian@MacBook-Pro-de-Sebastian P2 % python E5.py
To resize BBB video enter: 1, To change the codec to h264 enter 2
1
In what size do you want to resize you video?720/480/320/160
720
ffmpeg version N-99847-gcfddec Copyright (c) 2000-2020 the FFmpeg developers
  built with Apple clang version 12.0.0 (clang-1200.0.32.21)
  configuration: --prefix=/Users/sebastian/ffmpeg_build --pkg-config-flags=-
sebastian/ffmpeg_build/lib --extra-libs='-lpthread -lm' --bindir=/Users/sebastian
ble-libfreetype --enable-libbmp3lame --enable-libopus --enable-libsvtav1 --enable-
libavutil      56. 60.100 / 56. 60.100
libavcodec     58.112.101 / 58.112.101
The file has been saved in :BBBnew.avi
```

All the exercises are also on my repository on Github
<https://github.com/sebastianderson/P2.git>