

# Estudi previ

## Pràctica Àudio

Ernest Anguera Aixalà

Mariona Farré Tapias

Naïm Barba Morilla

Adrián Romera González

## Índex

Exercici 1 – Anàlisi d'arxius WAV .....	4
Apartat a .....	4
Apartat b .....	6
Apartat c .....	6
Apartat d .....	7
Apartat e .....	9
Exercici 2 – Mides d'arxiu .....	9
Apartat a .....	10
Apartat b .....	10
Exercici 3 – Codificació d'àudio en Telefonia IP .....	11
Apartat a .....	11
Apartat b .....	13
Exercici 4 – Càlculs MPEG .....	13
Apartat a .....	13
Apartat b .....	13
Apartat c .....	14
Exercici 5 – FFmpeg .....	14
Apartat a .....	14
Apartat b .....	15
Apartat c .....	16
Apartat d .....	16
Exercici 6 – Transmissió d'àudio sobre xarxes IP .....	17
Apartat a .....	17
Apartat b .....	17
Exercici 7 – Ràdios d'Internet .....	18
Apartat a .....	18
Apartat b .....	18
Apartat c .....	19
Exercici 8 – Còdecs utilitzats pels serveis de streaming i videoconferència .....	20
Exercici 9 – Planificació de les activitats de laboratori .....	22
Exercici 1 .....	22
Apartat a .....	22
Apartat b .....	22
Apartat c .....	22
Exercici 2 .....	22

Apartat a .....	22
Apartat b .....	22
Apartat c .....	22
Apartat d .....	23
Apartat e .....	23
Apartat f .....	23
Apartat g .....	23
Exercici 3 .....	23
Apartat a .....	23
Apartat b .....	23
Apartat c .....	23
Exercici 4 .....	23
Exercici 5 .....	24
Apartat a .....	24
Apartat b .....	24
Apartat c .....	24
Apartat d .....	24
Exercici 6 .....	24
Apartat a .....	24
Apartat b .....	24
Bibliografia .....	25

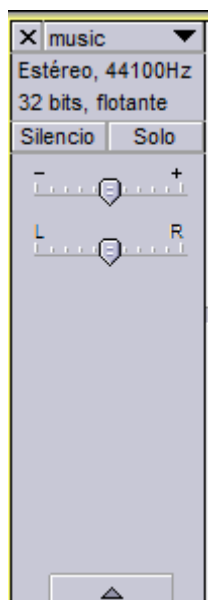
## Exercici 1 – Anàlisi d'arxius WAV

### Apartat a

Es pot observar que el fitxer *voice.wav* té so mono, això es pot saber, primer perquè s'escolta igual en els dos canals auditius i, segon, si observem el menú *Voice* que hi ha a la pista, se'ns informa que l'àudio està en mono:

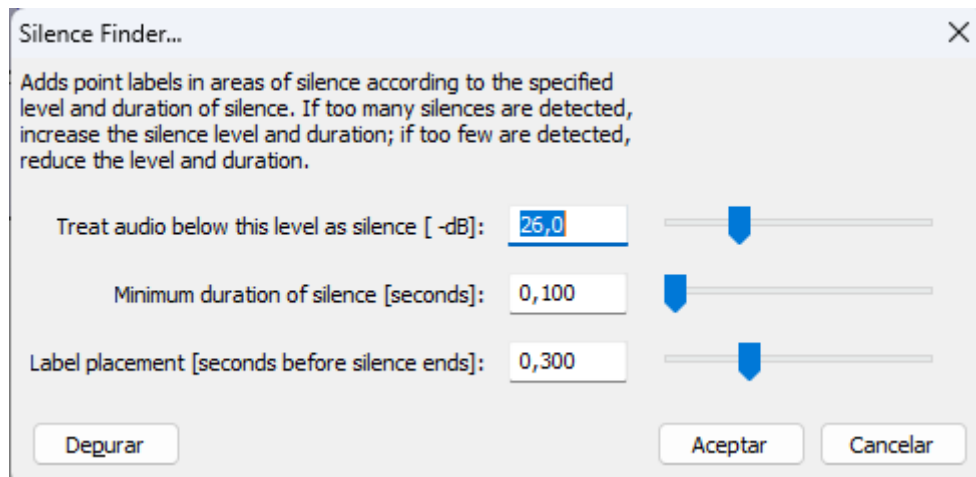


En canvi, la pista *music.wav*, està en estèreo, ja que s'escolta diferent a cada canal auditiu i si observem al menú *music* de la pista, se'ns informa que està en estèreo.



També podem veure que mentre que l'arxiu de veu està a una freqüència de 22050 Hz, l'arxiu de música funciona a 44100 Hz, aquesta diferència es fa present en la qualitat de l'àudio, ja que el so a la pista amb música és més ric que el de la pista de veu.

Podem detectar silencis de l'àudio realitzant un: Analizar → Silence Finder → Escollir Opcions (En aquest cas, s'ha provat amb silencis superiors a 0,1 segons):



En aquest punt, es pot veure una grandíssima diferència entre ambdues pistes, això es pot deure al fet que una pista conté les pauses entre “cançons”, mentre que la pista de veu, únicament conté una mostra en silenci.

A la pista de música, es pot observar que, amb la nostra configuració, hi ha 37 punts de silenci:



Si veiem amb més detall i ampliem la mostra, podem veure que la majoria d'aquests silencis es produeixen al principi, precisament quan no hi ha veu i dos instruments s'intercanvien. Aquests silencis es produeixen quan hi ha el canvi d'instrument. Els silencis al final de la pista són deguts a les pauses que fa la cantant entre versos.

## Apartat b

El format de fitxer WAV és un dels més senzills per emmagatzemar mostres d'àudio. A diferència dels formats comprimits com el MPEG, els fitxers WAV emmagatzemen les mostres "en brut" sense necessitat de preprocessament. Aquí teniu els detalls clau:

1. Estructura del fitxer WAV:
  - a. Chunk RIFF: Identifica el fitxer com a WAV. Conté informació sobre la longitud total del fitxer.
  - b. Chunk FORMAT: Defineix paràmetres com la taxa de mostreig i el nombre de canals (mono o estèreo).
  - c. Chunk DATA: Conté les dades reals (mostres) de l'àudio.
2. Exemple de capçalera de fitxer WAV:
  - a. DING.WAV és un fitxer WAV de 8 bits, mono, amb una taxa de mostreig de 22.050 Hz i una longitud d'11598 bytes.
  - b. La capçalera conté informació sobre la longitud del fitxer, el format (8 bits, mono), la taxa de mostreig i els bytes per mostra.
3. Exemple de fitxer WAV estèreo de 16 bits:
  - a. Taxa de mostreig: 44.100 Hz.
  - b. Bytes per segon: 4 vegades la taxa de mostreig (16 bits \* 2 canals).
  - c. Bits per mostra: 16.

En resum, els fitxers WAV són senzills i contenen informació essencial per reproduir àudio sense processament addicional.

## Apartat c

Voz.wav:

- a. Format d'àudio (AudioFormat): Per a un arxiu WAV sense comprimir, aquest camp seria 0x0001.
- b. Nombre de canals (NumChannels): Com que l'arxiu és mono, aquest camp seria 0x0001.
- c. Freqüència de mostreig (SampleRate): Com que la freqüència de mostreig és de 22050 Hz, aquest camp seria 0x00005622.
- d. Taxa de bits (Bitrate): Aquest camp es calcula com  $SampleRate * NumChannels * BitsPerSample$ .  
En aquest cas, seria:  $Bitrate = 22050 \cdot 1 \cdot 16 = 352800 \text{ bits/s}$   
El valor hexadecimal per a Bitrate seria 0x55F20.
- e. Profunditat de bits (BitsPerSample): Com que estàs utilitzant una representació de punt flotant de 16 bits, aquest camp seria 0x0010.

Voz-tel.wav:

- a. Format d'àudio (AudioFormat): Per un arxiu WAV sense comprimir, el valor d'aquest camp seria 0x0001.
- b. Nombre de canals (NumChannels): Com que l'arxiu és mono, aquest camp seria 0x0001.
- c. Freqüència de mostreig (SampleRate): Donat que la freqüència 8000 Hz, aquest camp seria 0x00001F40.

- d. Taxa de bits (Bitrate): Aquest camp es calcula com  $SampleRate * NumChannels * BitsPerSample$ .

Aquest cas, seria:  $BitRate = 8000 \cdot 1 \cdot 8 = 64000 \text{ bits/s}$

El valor hexadecimal pel BitRate seria 0xFA00.

- e. Profunditat de bits (BitsPerSample): Donat que s'està usant una representació de 8 bits PCM, aquest camp seria 0x0008.

Musica.wav:

- a. Format d'àudio (AudioFormat): Per a un arxiu WAV sense comprimir, aquest camp seria 0x0001.  
b. Nombre de canals (NumChannels): Com que l'arxiu és estèreo, aquest camp seria 0x0002.  
c. Freqüència de mostreig(SampleRate): Com que la freqüència de mostreig és de 44100 Hz, aquest camp seria 0x0000AC44.  
d. Taxa de bits (ByteRate): Aquest camp es calcula com  $SampleRate * NumChannels * BitsPerSample / 8$ . Així que en aquest cas seria:

$$ByteRate = 44100 * 2 * 16 = 1411200 \text{ bits/s}$$

Així que el valor hexadecimal per a ByteRate seria 0x158D80.

- e. Profunditat de bits (BitsPerSample): Com que estàs utilitzant una representació de punt flotant de 16 bits, aquest camp seria 0x0010.

## Apartat d

Voz.wav:

General	
Nom complet :	C:\Users\nesta\Downloads\Files\Files\voice.wav
Format :	Wave
Mida de l'arxiu :	56,7 KiB
Durada :	1 s 312 ms
Mode general velocitat bits :	Constant
Velocitat bits general :	354 kb/s
Codificat per :	David
Data d'enregistrament :	2006-05-02
Aplicació :	Sound Forge 4.5

Audio	
Format :	PCM
Ajustos del perfil, Endianness :	Little
Ajustos del perfil, SBR :	Signed
ID còdec :	1
Durada :	1 s 312 ms
Mode de taxa de bits :	Constant
Taxa de bits :	352,8 kb/s
Canal(s) :	1 canal
Freqüència de mostreig :	22,05 kHz
Profunditat de bits :	16 bits
Mida pista :	56,5 KiB (100%)

- Es pot observar que les dades obtingudes en realitzar els càlculs són les mateixes que les obtingudes en fer l'anàlisi amb MediaInfo.

Voz-tel.wav:

General	
Nom complet :	C:\Users\nesta\Downloads\Files\Files\voice-tel.wav
Format :	Wave
Mida de l'arxiu :	10,3 KiB
Durada :	1 s 312 ms
Mode general velocitat bits :	Constant
Velocitat bits general :	64,3 kb/s

Àudio	
Format :	PCM
Ajustos del perfil, Endianness :	Little
Ajustos del perfil, SBR :	Unsigned
ID còdec :	1
Durada :	1 s 312 ms
Mode de taxa de bits :	Constant
Taxa de bits :	64,0 kb/s
Canal(s) :	1 canal
Frequència de mostreig :	8 000 Hz
Profunditat de bits :	8 bits
Mida pista :	10,3 KiB (100%)

- Es pot veure que les dades aconseguides en fer els càlculs són les mateixes que les aconseguides en fer l'anàlisi amb MediaInfo.

Musica.wav:

General	
Nom complet :	C:\Users\nesta\Downloads\Files\Files\music.wav
Format :	Wave
Mida de l'arxiu :	11,4 MiB
Durada :	1 min 7 s
Mode general velocitat bits :	Constant
Velocitat bits general :	1 411 kb/s
Àlbum :	Chicago Soundtrack 2002
Nom de la pista :	Nowadays Hot Honey Rag Medley Title
Nom de la pista/Posició :	13
Intèrpret :	Chicago Soundtrack Featuring Renee Zellweger Catherine Zeta Jones & Taye Diggs
Director :	Chicago Soundtrack Featuring Renee Zellweger Catherine Zeta Jones & Taye Diggs
Gènere :	Sound Track
Data d'enregistrament :	2002
Formulari font original/Nom :	Chicago Soundtrack 2002
ITRK :	13

Àudio	
Format :	PCM
Ajustos del perfil, Endianness :	Little
Ajustos del perfil, SBR :	Signed
ID còdec :	1
Durada :	1 min 7 s
Mode de taxa de bits :	Constant
Taxa de bits :	1 411,2 kb/s
Canal(s) :	2 canals
Frequència de mostreig :	44,1 kHz
Profunditat de bits :	16 bits
Mida pista :	11,4 MiB (100%)

- Es pot veure que les dades assolides en fer els càlculs són les mateixes que les assolides en fer l'anàlisi amb MediaInfo.



## Apartat e

Voz.wav:

- *Bitrate*:  $22050 \text{ Hz} * 1 \text{ canal} * 16 \text{ bits} = 705600 \text{ bits/s} = 352,8 \text{ kbit/s}$
- *Mida arxiu*:  $352,8 \text{ kbit/s} * 1.312 \text{ segons de la pista} = 462873,6 \text{ bits} / 8 \text{ bits} = 57859,2 \text{ bytes} = 56,5 \text{ KiB}$

Si comparem els resultats obtinguts amb els proporcionats per MediaInfo, podem observar que amb les dades extretes amb Audacity, tenim el mateix bitrate i mida de l'arxiu.

Voz-tel.wav:

- *Bitrate*:  $8000 \text{ Hz} * 1 \text{ canal} * 8 \text{ bits} = 64000 \text{ bits/s} = 64 \text{ kbits/s}$
- *Mida arxiu*:  $64 \text{ kbits/s} * 1.312 \text{ segons de la pista} / 8 \text{ bits} = 10496 \text{ bytes} = 10,25 \text{ KiB}$

Si comparem els resultats aconseguits, podem veure que aconseguim el mateix bitrate que amb MediaInfo, en canvi, la mida de l'arxiu és lleugerament diferent, això es pot deure a arrodoniments.

Musica.wav:

- *Bitrate*:  $44100 \text{ Hz} * 2 \text{ canals} * 16 \text{ bits} = 1411200 \text{ bits/s} = 1,4112 \text{ Mbit/s}$
- *Mida arxiu*:  $1,4112 \text{ Mbit/s} * 67 \text{ segons de la pista} / 8 \text{ bits} = 11818800 \text{ bytes} = 11,27 \text{ MiB}$

Mentre que les dades de durada, metadades generals i la majoria de les especificacions de l'àudio coincideixen entre MediaInfo i els càlculs, hi ha una petita discrepància en la mida de l'arxiu, probablement deguda a la forma en què es redueixen les dades o s'arrodoneixen en els càlculs.

## Exercici 2 – Mides d'arxiu

Per calcular la mida del fitxer resultant quan el fitxer PCM cru es codifica amb diferents algorismes de compressió d'àudio, primer necessitem conèixer la mida original del fitxer PCM cru.

Atès que el fitxer PCM cru té les següents característiques:

- 8 bits/mostra
- Mostrejat a 22,05 kHz
- Mono
- 28948 mostres

Podem calcular la mida del fitxer original de la següent manera:

$$\text{Mida del fitxer original} = \text{Mida de mostra} * \text{Nombre de mostres}$$

$$\text{Mida de mostra} = \text{Bits per mostra} / 8 \text{ (per convertir bits a bytes)}$$

Per al fitxer PCM cru:

$$\text{Mida de mostra} = 8 / 8 = 1 \text{ byte}$$

$$\text{Nombre de mostres} = 28948$$

$$\text{Mida del fitxer original} = 1 \text{ byte/mostra} * 28948 \text{ mostres} = 28948 \text{ bytes}$$

Ara podem calcular la mida del fitxer resultant per a cada mètode de codificació d'àudio.

#### Apartat a

1. Adaptació Backward:

Per ADPCM, la taxa de bits és de 4 bits/mostra de diferència. Això vol dir que cada mostra es codifica utilitzant 4 bits menys que la mostra original.

$$\begin{aligned} \text{Mida del fitxer ADPCM} &= (\text{Mida del fitxer original} * 4) / 8 \\ \text{Mida del fitxer ADPCM (Adaptació Backward)} &= (28948 * 4) / 8 \\ &= 14474 \text{ bytes} \end{aligned}$$

2. Adaptació Forward amb predicció de 4a ordre actualitzada cada 25 ms:

Aquí, la taxa de bits continua sent de 4 bits/mostra de diferència, però es fa servir una tècnica de predicció més avançada.

La mida del fitxer resultant seria similar a l'anterior, ja que la taxa de bits és la mateixa, per la qual cosa també seria de 14.474 bytes.

#### Apartat b

1. A 4,75 kbit/s:

Primer, convertim la taxa de bits a bytes per segon (per ser consistent amb la mida del fitxer):

$$\begin{aligned} \text{Taxa de bits} &= 4,75 \text{ kbit/s} = 4750 \text{ bits/s} \\ \text{Taxa de bytes} &= \text{Taxa de bits} / 8 = 4750 / 8 = 593,75 \text{ bytes/s} \\ \text{Mida del fitxer AMR (4,75 kbit/s)} &= \text{Taxa de bytes} * \text{Durada de l'àudio (en segons)} \end{aligned}$$

Per trobar la durada de l'àudio en segons, dividim el nombre de mostres per la freqüència de mostreig:

$$\begin{aligned} \text{Durada de l'àudio (en segons)} &= \text{Nombre de mostres} / \text{Freqüència de mostreig} \\ \text{Durada de l'àudio (en segons)} &= 28948 / 22050 \approx 1.313 \text{ segons} \\ \text{Mida del fitxer AMR (4,75 kbit/s)} &= 593,75 \text{ bytes/s} * 1.313 \text{ s} \approx 779,56 \text{ bytes} \end{aligned}$$

2. A 12.2 kbit/s:

Seguint un procés similar a l'anterior:

$$\begin{aligned} \text{Taxa de bits} &= 12,2 \text{ kbit/s} = 12200 \text{ bits/s} \\ \text{Taxa de bytes} &= 12200 / 8 = 1525 \text{ bytes/s} \\ \text{Durada de l'àudio (en segons)} &= 28948 / 22050 \approx 1.313 \text{ segons} \\ \text{Mida del fitxer AMR (12,2 Kbit/s)} &= 1525 \text{ bytes/s} * 1.313 \text{ s} \approx 2001,13 \text{ bytes} \end{aligned}$$

Aleshores, resumint:

- Mida del fitxer ADPCM (Adaptació Backward): 14474 bytes
- Mida del fitxer ADPCM (Adaptació Forward): 14474 bytes
- Mida del fitxer Vocoder AMR (4,75 kbit/s): 780 bytes
- Mida del fitxer Vocoder AMR (12,2 kbit/s): 2001 bytes

## Exercici 3 – Codificació d'àudio en Telefonia IP

### Apartat a

Linphone10:

1. Opus:
  - Família: Perceptiu
  - Algoritme: CELT (Constrained Energy Lapped Transform)
  - Paràmetres rellevants: Amplada de banda adaptable (6-20 kHz), freqüència de mostreig (48 kHz), taxa (6-510 kbps)
2. Speex:
  - Família: Vocoders
  - Algoritme: CELP (Code Excited Linear Prediction)
  - Paràmetres rellevants: Amplada de banda (Estreta 8 kHz, Amplada 16 kHz), taxa (2,15-24,6 kbps)
3. PCMU/PCMA (G.711):
  - Família: PCM (Pulse Code Modulation)
  - Algoritme:  $\mu$ -law (PCMU), A-law (PCMA)
  - Paràmetres rellevants: Amplada de banda (300-3400 Hz), freqüència de mostreig (8 kHz), taxa (64 kbps)
4. GSM:
  - Família: Vocoders
  - Algoritme: RPE-LTP (Regular Pulse Excitation Long-Term Prediction)
  - Paràmetres rellevants: Amplada de banda (300-3400 Hz), freqüència de mostreig (8 kHz), taxa (13 kbps)
5. G722:
  - Família: Sub-Band ADPCM
  - Algoritme: SB-ADPCM (Sub-Band Adaptive Differential Pulse Code Modulation)
  - Paràmetres rellevants: Amplada de banda (50-7000 Hz), freqüència de mostreig (16 kHz), taxa (48/56/64 kbps)
6. iLBC:
  - Família: Vocoders
  - Algoritme: Block Independent Linear Predictive Coding
  - Paràmetres rellevants: Amplada de banda (300-3400 Hz), freqüència de mostreig (8 kHz), taxa (13,33 kbps per a trames de 30 ms, 15,20 kbps per a trames de 20 ms)
7. G729:
  - Família: Vocoders
  - Algoritme: CS-ACELP (Conjugate-Structure Algebraic-Code-Excited Linear-Prediction)
  - Paràmetres rellevants: Amplada de banda (300-3400 Hz), freqüència de mostreig (8 kHz), taxa (8 kbps)
8. BV16:
  - Família: Vocoders
  - Algoritme: BroadVoice16 (16 kbps estrets)
  - Paràmetres rellevants: Amplada de banda (200-3400 Hz), freqüència de mostreig (8 kHz), taxa (16 kbps)

9. L16:

- Família: PCM (Pulse Code Modulation)
- Algoritme: Linear16
- Paràmetres rellevants: Amplada de banda (Depèn de la freqüència de mostreig), freqüència de mostreig (44100 Hz, 32000 Hz), taxa (705,6 kbps, 1024 kbps)

Còdecs utilitzats en MicroSIP:

1. Opus:

- Família: Perceptiu
- Algoritme: CELT (Constrained Energy Lapped Transform)
- Paràmetres rellevants: Amplada de banda adaptable (6-20 kHz), freqüència de mostreig (48 kHz), taxa (6-510 kbps)

2. Speex:

- Família: Vocoders
- Algoritme: CELP (Code Excited Linear Prediction)
- Paràmetres rellevants: Amplada de banda (Estreta 8 kHz, Amplada 16 kHz), taxa (2,15-24,6 kbps)

3. PCMU/PCMA (G.711):

- Família: PCM (Pulse Code Modulation)
- Algoritme:  $\mu$ -law (PCMU), A-law (PCMA)
- Paràmetres rellevants: Amplada de banda (300-3400 Hz), freqüència de mostreig (8 kHz), taxa (64 kbps)

4. GSM:

- Família: Vocoders
- Algoritme: RPE-LTP (Regular Pulse Excitation Long-Term Prediction)
- Paràmetres rellevants: Amplada de banda (300-3400 Hz), freqüència de mostreig (8 kHz), taxa (13 kbps)

5. G722:

- Família: Sub-Band ADPCM
- Algoritme: SB-ADPCM (Sub-Band Adaptive Differential Pulse Code Modulation)
- Paràmetres rellevants: Amplada de banda (50-7000 Hz), freqüència de mostreig (16 kHz), taxa (48/56/64 kbps)

6. iLBC:

- Família: Vocoders
- Algoritme: Block Independent Linear Predictive Coding
- Paràmetres rellevants: Amplada de banda (300-3400 Hz), freqüència de mostreig (8 kHz), taxa (13,33 kbps per a trames de 30 ms, 15,20 kbps per a trames de 20 ms)

7. G729:

- Família: Vocoders
- Algoritme: CS-ACELP (Conjugate-Structure Algebraic-Code-Excited Linear-Prediction)
- Paràmetres rellevants: Amplada de banda (300-3400 Hz), freqüència de mostreig (8 kHz), taxa (8 kbps)

8. SILK:

- Família: Vocoders
- Algoritme: SILK
- Paràmetres rellevants: Amplada de banda (Estreta 8 kHz, Mitjana 12 kHz, Amplada 16 kHz, Superamplada 24 kHz), taxa (6-40 kbps)

#### 9. Linear PCM:

- Família: PCM (Pulse Code Modulation)
- Algoritme: Linear PCM
- Paràmetres rellevants: Amplada de banda (Depèn de la freqüència de mostreig), freqüència de mostreig (44100 Hz, 32000 Hz), taxa (705,6 kbps, 1024 kbps)

#### Apartat b

- Detecció de silenci (VAD, Voice Activity Detection): Aquest és un mecanisme que detecta la presència o absència de veu en el senyal d'àudio.

## Exercici 4 – Càlculs MPEG

#### Apartat a

En el cas de MPEG-1 Layer II (MP2) i Layer III (MP3), el nombre de mostres per trama és constant:

Per a MP2, una trama conté 1152 mostres.

Per a MP3, una trama també conté 1152 mostres.

Atès que la freqüència de mostreig del vostre fitxer és de 44,1 kHz (segons les dades de MediaInfo), podeu calcular el temps d'àudio per trama de la següent manera:

$$\text{Temps de trama} = \text{mostres per trama} / \text{freqüència de mostreig}$$

Per tant, per a MP2 i MP3:

$$\text{Temps de trama} = 1152 \text{ mostres} / 44100 \text{ mostres/segon} \approx 0.026 \text{ segons} = 26 \text{ ms}$$

Això indica que cada trama d'àudio en MP2 i MP3 representa 26 mil·lisegons (ms) d'àudio. Aquest valor és estàndard i està definit en les especificacions de MPEG.

#### Apartat b

El càlcul bàsicament és saber que cada trama són 26 ms i generem x kbps, multipliquem aquests valors, els dividim entre 8 i obtenim el resultat. Podem verificar aquesta informació amb mpeg audioinfo.

32mp2 trames de 104 o 105 bytes, les de 105 tenen padding

96mp2 trames de 313 o 314 bytes, les de 314 tenen padding

128mp2 trames de 417 o 418 bytes, les de 418 tenen padding

32mp3 trames de 104 o 105 bytes, les de 105 tenen padding

96mp3 trames de 313 o 314 bytes, les de 314 tenen padding

128.mp3 trames de 417 o 418 bytes, les de 418 tenen padding

El padding s'utilitza per fer que la mida de les trames sigui un nombre enter de bytes. En aquest cas, les trames amb padding són 1 byte més gran que les trames sense padding. Això es deu al fet que la mida de les trames ha de ser un nombre enter de bytes, i el padding s'usa per omplir l'espai extra quan la mida de la trama no és un nombre enter.

### Apartat c

El nombre total de trames es pot calcular dividint la durada total de l'àudio (en ms) pel temps de trama (26 ms per a MP2 i MP3). La durada del vostre àudio és d'1 minut i 7 segons, que és igual a 67000 ms. Per tant, el nombre total de trames seria:

$$\text{Nombre total de trames} = 67000 \text{ ms} / 26 \text{ ms/trama} \approx 2577 \text{ trames}$$

La mida total (bytes) del fitxer comprimit és simplement el nombre de trames multiplicat per la mida de la trama. En conseqüència, per a cada bitrate i format, tindríem:

Per a 32 kbit/s MP2 i MP3, la mida total seria:

$$2577 \text{ trames} \times 104 \text{ bytes/trama} = 267608 \text{ bytes} = 261 \text{ KiB}$$

Per a 96 kbit/s MP2 i MP3, la mida total seria:

$$2577 \text{ trames} \times 313 \text{ bytes/trama} = 806081 \text{ bytes} = 787 \text{ KiB}$$

Per a 128 kbit/s MP2 i MP3, la mida total seria:

$$2577 \text{ trames} \times 417 \text{ bytes/trama} = 1074099 \text{ bytes} = 1048 \text{ KiB}$$

## Exercici 5 – FFmpeg

### Apartat a

En la documentació donada, podem trobar informació rellevant com es codifiquen els arxius .wav utilitzant l'eina de FFmpeg.

Per poder trobar aquesta informació, hem d'anar a la web oficial de FFmpeg i entrar en la seva documentació: <https://ffmpeg.org/documentation.html>

Dins d'aquesta per poder trobar informació de la conversió d'arxius, anar a la pàgina sota la secció de: "Command Line Tools Documentation" i seleccionar la documentació de: ffmpeg, <https://ffmpeg.org/ffmpeg.html>

En aquesta pàgina hi ha tota la informació relacionada amb, sota l'apartat 6.3 Video and Audio file format conversion, trobem la informació que necessitem: <https://ffmpeg.org/ffmpeg.html#toc-Video-and-Audio-file-format-conversion>

Dins d'aquesta hi ha informació i diversos exemples de comandes ffmpeg per a diferents operacions de conversió i manipulació de fitxers multimèdia, per al propòsit de codificar arxius WAV d'entrada a qualsevol codificador de sortida.

El comandament que es necessita per aquesta conversió d'un arxiu d'àudio WAV a un format específic usant ffmpeg, la comanda general podria seguir el fil de:

```
ffmpeg -i arxiuentrada.wav -opcions per la codificació  
arxiusortida.mp2
```

I un exemple seria:

```
ffmpeg -i /tmp/a.wav -ar 22050 /tmp/a.mp2
```

Amb aquesta comanda -i /tmp/a.wav especifica l'arxiu WAV d'entrada "a.wav", i -ar 22050 estableix la freqüència de mostreig d'àudio a 22050 Hz. Després, /tmp/a.mp2 especifica l'arxiu de sortida, que es convertirà al format MP2.

Aquesta comanda es pot adaptar a qualsevol arxiu de sortida canviant els paràmetres segons el còdec i la configuració que vulgueu utilitzar per a la sortida. Per exemple, si voleu convertir a un format MP3, podeu canviar l'extensió de l'arxiu de sortida a .mp3 i, opcionalment, ajustar altres paràmetres com la taxa de bits d'àudio, la freqüència de mostreig entre molts altres.

### Apartat b

Ara entenem com es poden codificar els arxius de sortida a qualsevol còdec, ho podem aplicar als diferents tipus d'arxiu.

- MP3 amb bitrate constant, CBR

En la codificació del còdec MP3 amb CBR, el bitrate de l'arxiu de sortida s'ha de mantenir constant durant tota la reproducció. Això significa que cada segon de l'arxiu MP3 contindrà aproximadament la mateixa quantitat d'informació de so.

La comanda general seria:

```
ffmpeg -i arxiuentrada.wav -c:a libmp3lame -b:a bitrate  
arxiusortida.mp3
```

Amb aquesta comanda, s'utilitza ffmpeg per la codificació, amb "-i arxiuentrada.wav", especificant l'arxiu .wav d'entrada. L'opció "-c:a libmp3lame" indica que s'usa el còdec libmp3lame per a la codificació d'àudio MP3, seguida de l'opció de "-b:a bitrate" és per especificar quin bitrate es desitja per a la codificació, en kilobits per segon (kbps), al final es genera l'arxiu mp3 de sortida amb el nom "arxiusortida.mp3".

Un exemple podria ser:

```
ffmpeg -i musica.wav -c:a libmp3lame -b:a 128k musicasortida.mp3
```

Amb aquest exemple musica.wav és l'arxiu d'entrada en format .wav. S'està fent servir el còdec libmp3lame per a la codificació MP3, i s'està especificant un bitrate de 128 kbps (-b:a 128k) i l'arxiu de sortida serà musicasortida.mp3.

- MP3 amb bitrate variable, VBR

En la codificació del còdec MP3 amb VBR, el bitrate de l'arxiu de sortida varia en funció de la complexitat del contingut d'àudio. En les parts més simples o silencioses de la música, el bitrate serà menor, mentre que en les parts més complexes i detallades, el bitrate serà major.

La comanda general seria:

```
ffmpeg -i arxiuentrada.wav -c:a libmp3lame -q:a qualitat  
arxiusortida.mp3
```

Amb aquesta comanda, s'utilitza ffmpeg per la codificació, amb "-i arxiuentrada.wav", especificant l'arxiu .wav d'entrada. L'opció "-c:a libmp3lame" indica que s'usa el còdec libmp3lame per a la codificació d'àudio MP3, seguida de l'opció de "-q:a qualitat" és per especificar quina qualitat es vol determinar l'objectiu de la taxa de bits, on un valor més baix produeix una millor qualitat però arxius més grans, i un valor més alt produeix una qualitat menor però arxius més petits. Al final es genera l'arxiu MP3 de sortida amb el nom "arxiusortida.mp3".

Un exemple podria ser:

```
ffmpeg -i musica.wav -c:a libmp3lame -q:a 5 musicasortida.mp3
```

Amb aquest exemple musica.wav és l'arxiu d'entrada en format .wav. S'està utilitzant el còdec libmp3lame per a la codificació MP3, i s'està especificant una qualitat de codificació de 5 (-q:a 5). Aquesta qualitat determina la compressió de l'àudio, on valors més baixos indiquen millor qualitat, però arxius més grans, l'arxiu de sortida serà musicasortida.mp3.

### Apartat c

Per descodificar des de qualsevol còdec cap al còdec de tipus .wav, utilitzarem el mateix principi, de les anteriors preguntes, però invertint els paràmetres.

La comanda general seria:

```
ffmpeg -i arxiuentrada.mp3 -c:a pcm_s16le arxiusortida.wav
```

Amb aquesta comanda, s'usa ffmpeg per la codificació, amb "-i arxiuentrada.wav", especificant l'arxiu .wav d'entrada. L'opció "-c:a pcm\_s16le" indica que es fa servir el còdec "pcm\_s16le" per la sortida d'un arxiu amb una codificació d'àudio pcm wav. Al final es genera l'arxiu .wav de sortida amb el nom "arxiusortida.wav".

Un exemple podria ser:

```
ffmpeg -i musica.mp3 -c:a pcm_s16le -ar 48000 musicasortida.wav
```

Amb aquest exemple musica.mp3 és l'arxiu d'entrada en format .mp3. Utilitza el còdec pcm\_s16le per a la codificació de l'àudio de sortida en format PCM de 16 bits, petit endian, que és el format de .wav. Continuat amb l'opció: "-ar 48000" que especifica la freqüència de mostreig de l'àudio de sortida com a 48000 Hz, establint la qualitat de l'àudio i la freqüència de mostreig per a l'arxiu WAV de sortida, i l'arxiu de sortida serà musicasortida.wav.

### Apartat d

```
ffmpeg -i C:\music.wav -ac 2 -acodec libmp3lame -f mp3 -aq 5 -ar 44100 C:\musicvbr.mp3
```

Per entendre la comanda donada, dividir per les diferents opcions donades:

- -i C:\music.wav: Especifica l'arxiu d'entrada, en aquest cas, "C:\music.wav".
- -ac 2: Estableix el nombre de canals de sortida, en aquest es defineixen 2 indicant que és estèreo.
- -acodec libmp3lame -f mp3: Indica el còdec d'àudio a utilitzar per a la codificació, s'utilitza libmp3lame, que és el còdec d'MP3, i -f mp3 és per especificar el format de sortida com a MP3.
- -aq 5: Indica la qualitat de la codificació per a l'MP3, com amb l'MP3 VBR, com més baixa sigui el valor, millor serà la qualitat. En aquest cas, 5 és un valor mitjà de qualitat.
- -ar 44100: Estableix la freqüència de mostreig de l'àudio en Hz, que es dona que és 44100 Hz.
- C:\musicvbr.mp3: Especifica el nom de l'arxiu de sortida, que serà "C:\musicvbr.mp3".



```
ffmpeg -i C:\music.wav -vn -ac 2 -acodec mp2 -f mp2 -ab 128000 -ar 44100 C:\audio128kb.mp2
```

Per entendre la comanda donada, dividir per les diferents opcions donades:

- `-i C:\music.wav`: Especifica l'arxiu d'entrada, en aquest cas, "C:\music.wav".
- `-vn`: Deshabilita la sortida de vídeo, indicant a FFmpeg que no ha de generar arxius de vídeo per la sortida.
- `-ac 2`: Estableix el nombre de canals de sortida, en aquest es defineixen 2 indicant que és estèreo.
- `-acodec mp2 -f mp2`: Indica el còdec d'àudio a utilitzar per a la codificació, aquest utilitza MP2, que és el còdec d'àudio MPEG-1 Audio Layer II i `-f mp2` és per especificar el format de sortida com a MP2.
- `-ab 128000`: Estableix la taxa de bits d'àudio per al MP2 en bits per segon, aquest el fixa en 128000 bits/s, sent uns 128 kbit/s.
- `-ar 44100`: Estableix la freqüència de mostreig de l'àudio en Hz, que es dona que és 44100 Hz.
- `C:\audio128kb.mp2`: Especifica el nom de l'arxiu de sortida, que serà "C:\audio128kb.mp2"

## Exercici 6 – Transmissió d'àudio sobre xarxes IP

### Apartat a

Dins de la web de live555, entrar en la documentació de "Streaming Media" per trobar informació dels seus còdec: <http://www.live555.com/liveMedia/>

En l'apartat de testProgs, es pot veure la llista de totes les aplicacions per utilitzar les llibreries que codifiquen els còdec que pot processar són:

- MPEG-1 o 2: Àudio o vídeo, incloent-hi àudio MP3.
- MPEG-4: Vídeo
- H.264: Vídeo
- H.265: Vídeo
- H.263+: Vídeo
- JPEG: Imatges
- MPEG, incloent-hi arxius VOB, pel transport o fluxos de programes.
- Vídeo DV
- Àudio AMR
- Àudio WAV amb PCM
- Matroska o fitxers WebM

### Apartat b

En la descripció inicial de l'aplicació i el mateix apartat de testProgs es troba la informació sobre els protocols IP utilitzats per al transport i la senyalització, com menciona en la descripció:

"They can easily be extended to support additional (audio and/or video) codecs, and can also be used to build basic RTSP or SIP clients and servers,"

Utilitza per al transport els protocols RTP (Real-time Transport Protocol), per la transmissió en temps real de dades siguin d'àudio o vídeo a través de la xarxa.

I amb el protocol addicional de RTSP (Real Time Streaming Protocol), per poder controlar les sessions de transmissió de dades entre servidors i clients.

També per la senyalització de les sessions de transmissió, s'utilitza el protocol SIP (Session Initiation Protocol), que s'encarrega d'iniciar, modificar i finalitzar aquestes sessions entre els diferents servidors i clients.

Informació sobre el protocol RTP: <https://hop.extrahop.com/resources/protocols/rtp/>

Informació sobre el protocol RTSP: <https://www.wowza.com/blog/rtsp-the-real-time-streaming-protocol-explained>

Informació sobre el protocol SIP: <https://hop.extrahop.com/resources/protocols/sip>

## Exercici 7 – Ràdios d'Internet

### Apartat a

Els còdecs utilitzats són MP3 i AAC. Els bitrates típics són de 128 kbit/s, tot i que és habitual trobar emissions amb bitrates superiors com 192, i amb bitrates inferiors com 64, a més d'altres com 96 o 320. Les emissions amb un bitrate tenen qualitat molt bona, mentre que a mesura que es redueix el bitrate, es nota una baixada de qualitat.

### Apartat b

No.	Time	Source	Destination	Protocol	Length	Info
116	0.772132	192.168.1.100	194.97.151.149	TCP	66	57581 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
130	0.814289	194.97.151.149	192.168.1.100	TCP	66	80 → 57581 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM WS=512
131	0.814335	192.168.1.100	194.97.151.149	TCP	54	57581 → 80 [ACK] Seq=1 Ack=1 Win=262656 Len=0
132	0.814579	192.168.1.100	194.97.151.149	HTTP	562	GET /antenne?lcy=https HTTP/1.1
133	0.857928	194.97.151.149	192.168.1.100	TCP	60	80 → 57581 [ACK] Seq=1 Ack=509 Win=30720 Len=0
135	0.968928	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=1 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
136	0.968928	194.97.151.149	192.168.1.100	TCP	210	80 → 57581 [PSH, ACK] Seq=1461 Ack=509 Win=30720 Len=156 [TCP segment of a reassembled PDU]
137	0.968928	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=1617 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
138	0.968928	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=3077 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
139	0.968928	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=4537 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
140	0.968992	192.168.1.100	194.97.151.149	TCP	54	57581 → 80 [ACK] Seq=509 Ack=5997 Win=262656 Len=0
141	0.969014	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=5997 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
142	0.969014	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=7457 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
143	0.969014	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=8917 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
144	0.969014	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=10377 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
145	0.969014	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=11837 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
146	0.969029	192.168.1.100	194.97.151.149	TCP	54	57581 → 80 [ACK] Seq=509 Ack=13297 Win=262656 Len=0
147	0.998967	192.168.1.100	194.97.151.149	TCP	54	57581 → 80 [FIN, ACK] Seq=509 Ack=13297 Win=262656 Len=0
148	1.009384	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=13297 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
149	1.009384	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=14757 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
150	1.009384	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=16217 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
151	1.009506	192.168.1.100	194.97.151.149	TCP	54	57581 → 80 [RST, ACK] Seq=510 Ack=17677 Win=0 Len=0
152	1.009571	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=17677 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
153	1.009571	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=19137 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
154	1.009571	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=20597 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
155	1.009571	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=22057 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
156	1.009571	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=23517 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
157	1.009571	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=24977 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
158	1.009571	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=26437 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
159	1.009571	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=27897 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
160	1.009571	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=29357 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
161	1.009633	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=30817 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
162	1.009633	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=32277 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
163	1.009655	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=33737 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
164	1.009655	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=35197 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
165	1.009772	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=36657 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
166	1.009772	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=38117 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
167	1.009772	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=39577 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]
168	1.009803	194.97.151.149	192.168.1.100	TCP	1514	80 → 57581 [ACK] Seq=41037 Ack=509 Win=30720 Len=1460 [TCP segment of a reassembled PDU]

Frame 152: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits) on interface \Device\NPF\_{6770FFB9-578B-4EC5-A158-7BD72BA85049}, id 0

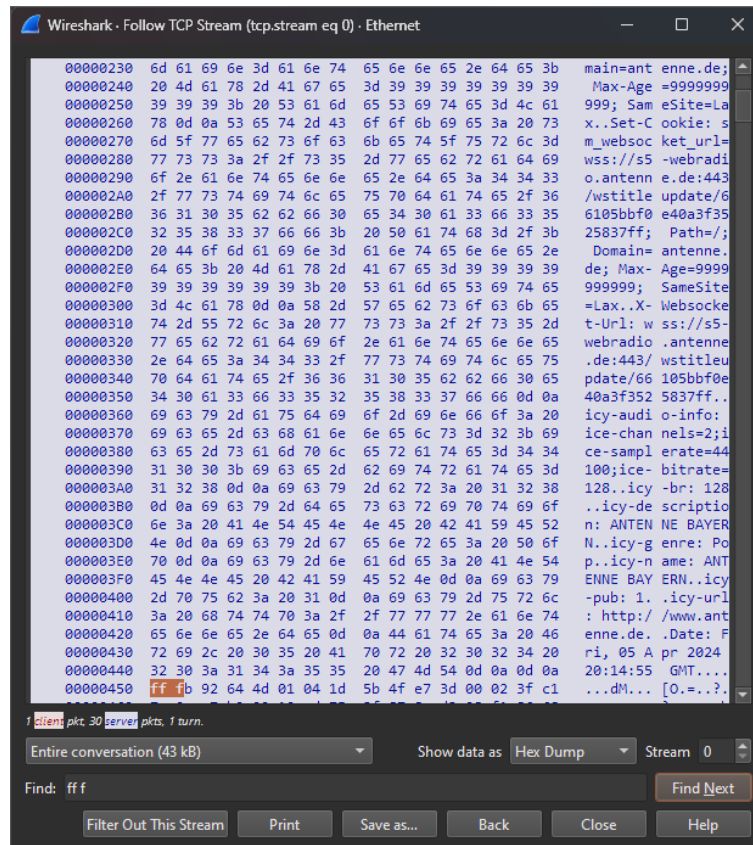
Ethernet II, Src: Arcadyan, f8:12:b7 (60:8d:26:f8:12:b7), Dst: Hewlett-Packard, 41:60:97 (c8:d3:ff:41:60:97)

Internet Protocol Version 4, Src: 194.97.151.149, Dst: 192.168.1.100

Transmission Control Protocol, Src Port: 80, Dst Port: 57581, Seq: 17677, Ack: 509, Len: 1460

S'ha utilitzat TCP per a la transmissió. La pila de protocols és TCP + IP + Ethernet. Dins del payload del segment TCP, es troba la capçalera d'àudio MPEG.

Apartat c



La capçalera d'àudio MPEG en hexadecimal és FFB9264. La descodificació és la següent:

- FFF = 111111111111, indica sincronització amb 12 bits a 1.
- B = 1011, on 1 és l'ID per MPEG-1, 01 és la capa 3, i l'últim bit a 1, indica absència de CRC.
- 9 = 1001, correspon a una taxa de bits de 128 kbit/s per ser MPEG-1 Capa 3.
- 2 = 0010, indica una freqüència de mostreig de 44,1 kHz per ser MPEG-1, sense padding i camp privat lliure.
- 6 = 0110, mostra el mode Joint estèreo i una extensió del mode.
- 4 = 0100, sense copyright, indica que és original i amb èmfasis a 0.

## Exercici 8 – Còdecs utilitzats pels serveis de streaming i videoconferència

- Videoconferència
  - Skype [1] [2]
    - Àudio: Opus
      - 24 kbit/s – 128 kbit/s
    - Vídeo: H.264
      - 128 kbit/s – 1,5 Mbit/s
  - Google Meet [3]
    - Àudio: Opus
    - Vídeo: VP8, VP9
  - Zoom [4]
    - Àudio: G.711, G.722, Opus
    - Vídeo: H.264, H.264 High Profile
- Streaming d'àudio
  - Spotify [5]
    - HE-AAC v2: 24 kbit/s
    - Vorbis: 96 kbit/s, 160 kbit/s, 320 kbit/s
    - AAC: 128 kbit/s, 256 kbit/s
  - SoundCloud [6]
    - AAC: 256 kbit/s
    - WAV
    - FLAC
    - AIFF
    - ALAC
  - Apple Music [7]
    - HE-AAC: 256 kbit/s
    - AAC: 256 kbit/s
    - ALAC:
      - 16-bit/44,1 kHz
      - 24-bit/48 kHz
      - 16-bit/88,2 kHz
      - 24-bit/88,2 kHz
      - 16-bit/192 kHz
      - 24-bit/192 kHz

- Amazon Music [8]
  - Opus:
    - Profunditat de bits: 16 bits
    - Taxa de mostreig: 48 kHz
    - Amplades de banda:
      - 48 kbit/s
      - 192 kbit/s
      - 320 kbit/s
  - FLAC:
    - Profunditat de bits: 16 bits (per a una variant) / 24 bits (per a altres variants)
    - Taxa de mostreig: 44,1 kHz (per a una variant) / 48 kHz, 96 kHz, 192 kHz (opcional per a altres variants)
    - Amplades de banda:
      - 800 kbit/s (mitjana)
      - 1600 kbit/s (mitjana)
      - 2800 kbit/s (mitjana)
      - 5000 kbit/s (mitjana)
  - Dolby Digital Plus Joint Object Coding (DD+JOC / EAC3-JOC):
    - Profunditat de bits: 16 bits
    - Taxa de mostreig: 48 kHz
    - Amplades de banda:
      - 384 kbit/s (6 canals - no Atmos)
      - 448 kbit/s (16 objectes base)
      - 768 kbit/s (16 objectes base)
  - MPEG-H (MHA1):
    - Profunditat de bits: 16 bits
    - Taxa de mostreig: 48 kHz
    - Amplades de banda:
      - 320 kbit/s (8 canals, 7.1)
      - 640 kbit/s (12 canals, 7.1.4)
      - 1024 kbit/s (12 canals, 7.1.4)
      - 1536 kbit/s (24 canals, 22.2)

## Exercici 9 – Planificació de les activitats de laboratori

### Exercici 1

#### Apartat a

- Obrim Audacity.
- Anem a "Fitxer" -> "Importa" -> "Dades en brut".
- Seleccionem l'arxiu PCM: voice8bit.sss.
- A la finestra que apareix, seleccionem els paràmetres correctes:
  - Codificació: Unsigned 8-bit PCM
  - Ordre de byte: Little-endian
  - Canals: 1 Canal (mono)
  - Freqüència de mostreig: 22050 Hz
- Premem "Importar".

#### Apartat b

- Observem la forma d'ona i la dividim en segments que corresponguin a fonemes.
- Calculem la durada típica dels fonemes.
- Analitzem l'espectrograma per identificar les freqüències dels fonemes.

#### Apartat c

- Repetim els passos de la secció a) amb paràmetres incorrectes, com per exemple:
  - Codificació: Unsigned 8-bit PCM
  - Ordre de byte: Little-endian
  - Canals: 1 Canal (mono)
  - Freqüència de mostreig: 44100 Hz
- Observem la forma d'ona i el so per veure com canvien.

Provem també de canviar la freqüència de mostreig a 16000 Hz.

### Exercici 2

#### Apartat a

- Codifiquem els arxius d'àudio segons la taula proporcionada.
- Per comparar en igualtat de condicions, utilitzem la mateixa qualitat original per a tots els arxius.
- Guardem els arxius codificats
  - Anem a "Fitxer" -> "Exportar àudio".
  - A la finestra que apareix, seleccionem els paràmetres correctes:
    - Format: (Fitxers FLAC, Fitxers Opus, Altres fitxers sense comprimir, Fitxers AMR (banda estreta) (FFmpeg))
  - Premem "Exportar".

#### Apartat b

- Per a alguns casos rellevants (almenys un per a música i un altre per a veu), representem les formes d'ona del fitxer comprimit juntament amb el WAV original.
- Alineem les formes d'ona en el temps.
- Descrivim les degradacions introduïdes pel còdec.

#### Apartat c

- Escoltem els arxius originals i codificats per avaluar la qualitat subjectiva.

- Comparem la qualitat de la música i la veu amb diferents codificadors.

#### Apartat d

- Comparem els nostres resultats amb estudis previs sobre la qualitat dels codificadors.
- Identifiquem possibles inconsistències i les expliquem.

#### Apartat e

- Identifiquem els còdecs que no introdueixen pèrdues i els que sí.
- Relacionem això amb els factors de compressió i la qualitat.

#### Apartat f

- Expliquem per què el concepte de bits/mostra no s'aplica en codificadors i compressors perceptius.

#### Apartat g

- Expliquem per què es pot configurar FLAC a diferents nivells.
- Definim a què fan referència aquests nivells i per què se'ls denomina "fastest" i "best".
- Codifiquem la veu amb AMR al bitrate més alt i al més baix.
- Comparem la qualitat entre els dos extrems.

### Exercici 3

#### Apartat a

- Codifiquem l'arxiu musica.wav amb els còdecs MP2 i MP3 d'Audacity.
  - Anem a "Fitxer" -> "Exportar àudio".
  - A la finestra que apareix, seleccionem els paràmetres correctes:
    - Format: (Arxius MP2, Arxius MP3)
  - Premem "Exportar".
- Codifiquem el mateix arxiu amb FFmpeg a través de la línia d'ordres.
  - `ffmpeg -i <fitxer> -vn -ac 2 -acodec mp2 -f mp2 -ab <freqüència_mostreig> -ar <bitrate> <fitxer_sortida>.mp2`
  - `ffmpeg -i <fitxer> -vn -ac 2 -acodec libmp3lame -f mp3 -ab <freqüència_mostreig> -ar <bitrate> <fitxer_sortida>.mp3`
- Omplim la taula amb els resultats.

#### Apartat b

- Comparem els resultats obtinguts amb els tres mètodes.
- Discutim la relació entre la compressió, el codificador utilitzat i la qualitat subjectiva.
- Resolem possibles discrepàncies amb els càlculs previs.

#### Apartat c

- Comparem la qualitat dels arxius codificats amb Audacity i FFmpeg per als mateixos còdecs i bitrates.
- Identifiquem possibles causes de les diferències (configuració per defecte dels còdecs, etc.).

### Exercici 4

- Observem les formes d'ona dels MP3 comprimits a 32 i 128 kbit/s per a mostres d'àudio de veu i música.
- Comparem els efectes de la compressió MPEG en aquests dos tipus d'àudio.
- Comentem si hi ha diferències significatives en el nivell de detall de la forma d'ona.

## Exercici 5

### Apartat a

- Explorem les opcions de codificació VBR disponibles a Audacity.
  - Anem a "Fitxer" -> "Exportar àudio".
  - A la finestra que apareix, seleccionem els paràmetres correctes:
    - Format: Arxius MP3
    - Qualitat: ?
  - Premem "Exportar".
- Comentem les característiques i limitacions d'aquestes opcions.

### Apartat b

- Justifiquem l'elecció dels paràmetres VBR a utilitzar.
- Es recomana triar una taxa de bits mitjana baixa per observar els efectes de la compressió MPEG.

### Apartat c

- Codifiquem l'arxiu musica.wav en format MP3 VBR amb els paràmetres seleccionats.
  - Anem a "Fitxer" -> "Exportar àudio".
  - A la finestra que apareix, seleccionem els paràmetres correctes:
    - Format: Arxius MP3
    - Qualitat: ?
  - Premem "Exportar".
- Codifiquem el mateix arxiu en format MP3 CBR amb la mateixa taxa de bits mitjana.
- Comparem la qualitat de les dues codificacions.

### Apartat d

- Utilitzem AnaMP3 i Excel per fer la gràfica l'evolució temporal de la taxa de bits de l'arxiu VBR.
- Extraïem conclusions sobre la relació entre la taxa de bits variable i les característiques de l'àudio.

## Exercici 6

- Configurem el servidor Live555.
- Configurem VideoLAN com a client RTSP en una altra màquina (o en un telèfon mòbil).
- Transmetem un arxiu de cada tipus: WAV (PCM), MP3 CBR i MP3 VBR.
- Capturem alguns paquets RTP amb Wireshark durant la transmissió.

### Apartat a

- Analitzem els camps RTP: increments de timestamp, tipus de càrrega i bit de marcador.

### Apartat b

- Identifiquem la pila de protocols i les regles de fragmentació.
- Calculem el nombre de mostres/trames d'àudio per paquet RTP i el temps d'àudio per paquet.
- Estimem el bitrate a nivell IP.



## Bibliografia

- [1] «Skype,» [En línia]. Disponible: <https://en.wikipedia.org/wiki/Skype>. [Últim accés: 4 Abril 2024].
- [2] «How much bandwidth does Skype need? | Skype Support,» [En línia]. Disponible: <https://support.skype.com/en/faq/FA1417/how-much-bandwidth-does-skype-need>. [Últim accés: 4 Abril 2024].
- [3] «Google Meet,» [En línia]. Disponible: [https://en.wikipedia.org/wiki/Google\\_Meet](https://en.wikipedia.org/wiki/Google_Meet). [Últim accés: 4 Abril 2024].
- [4] «Conference Room Connector supported devices,» [En línia]. Disponible: [https://support.zoom.com/hc/en/article?id=zm\\_kb&sysparm\\_article=KB0061702](https://support.zoom.com/hc/en/article?id=zm_kb&sysparm_article=KB0061702). [Últim accés: 4 Abril 2024].
- [5] «Calidad del audio,» [En línia]. Disponible: <https://support.spotify.com/es-gl/article/audio-quality/>. [Últim accés: 4 Abril 2024].
- [6] «Uploading requirements,» [En línia]. Disponible: <https://help.soundcloud.com/hc/en-us/articles/115003452847-Uploading-requirements>. [Últim accés: 4 Abril 2024].
- [7] «Apple Music,» [En línia]. Disponible: [https://en.wikipedia.org/wiki/Apple\\_Music](https://en.wikipedia.org/wiki/Apple_Music). [Últim accés: 4 Abril 2024].
- [8] «Audio Formats | Amazon Music Developer,» [En línia]. Disponible: <https://developer.amazon.com/de/docs/music/audio-formats.html>. [Últim accés: 4 Abril 2024].