

Tarea 7 - Programación y algoritmos
Giovanni Gamaliel López Padilla

Índice

1. Análisis de la complejidad de Quicksort	2
2. Encriptado y desencriptado de un archivo wav	2
2.1. Organización de la cabecera	2
2.2. Lectura y organización de los datos	3
2.3. Encriptación y desencriptación	3
2.3.1. Mapa caótico	3
2.3.2. Algoritmo	4
3. Referencias	4

1. Análisis de la complejidad de Quicksort

2. Encriptado y desencriptado de un archivo wav

2.1. Organización de la cabecera

Los datos de un archivo wav se encuentran administrados como header y los datos de audio como se muestra en la tabla 1.

Sección	Tamaño (bytes)	Descripción
RIFF	4	Contiene las letras R, I, F y F
	4	Entero positivo que almacena el tamaño de los bytes restantes
WAVE	4	Contienen las letras: W, A, V y E.
	4	Contiene los caracteres F, M, T
	4	Entero positivo que indica el tamaño en bytes del resto del bloque
	2	Entero positivo de 16 bits que indica el tipo de grabación. Un 1 significa PCM.
	2	Número de canales
	4	Frecuencia de muestreo expresada en Hz
	4	Número promedio de bytes por segundo
	2	Número de bytes usados en el archivo para cada muestra
	2	Bits por muestra
data	4	Contiene las letras d, a, t, a
	4	Entero positivo que indica el espacio de bytes que ocupan los datos
	n	Datos

Tabla 1: Organización de los datos en un archivo de formato wav.

Esta organización hace que los primeros 44 bytes no sean modificados, ya que es información del archivo y no sus datos. Al ser los archivos de muestra de un audio mono, entonces estos tiene un tamaño de 16 bits por muestra, esto se puede ver en las figuras 1 y 2.

Información del archivo ftea_10k.wav	
ChunkID	RIFF
ChunkSize	39060
Format	WAVE
Subchunk1ID	fmt
Subchunk1Size	16
AudioFormat	1
NumChannels	1
SampleRate	10000
ByteRate	20000
BlockAlign	2
BitsPerSample	16
blockID	data
blockSize	39024

Figura 1: Información del archivo [ftea_10k.wav](#)

Información del archivo mtea_10k.wav	
ChunkID	RIFF
ChunkSize	53054
Format	WAVE
Subchunk1ID	fmt
Subchunk1Size	16
AudioFormat	1
NumChannels	1
SampleRate	10000
ByteRate	20000
BlockAlign	2
BitsPerSample	16
blockID	data
blockSize	53018

Figura 2: Información del archivo [mtea_10k.wav](#)

Si estos fueran un audio stereo, entonces el número de bits por muestra sería 32, 16 para el audio izquierdo y 16 para el derecho. Es por ello que se implementó un arreglo de dimensión dinámica para guardar cada muestra del archivo.

2.2. Lectura y organización de los datos

La lectura de los datos de la cabecera serán guardados en tipos definidos. Estos tipos se encuentran en el archivo [wave.h](#). Este proceso no es propio, se encontró un repositorio en GitHub, el cual realiza modificaciones al sonido y se implementó el uso de sus lecturas de los datos. Estos repositorios son los siguientes: [Lectura de los datos](#), [Estructura de los tipos](#).

El arreglo de datos es de tipo short para que estos coincidan con los 16 bits que tiene cada muestra de audio. Al tener las muestras de audio organizadas en arreglos es más sencillo realizar el paso por cada uno de estos datos.

2.3. Encriptación y desencriptación

2.3.1. Mapa caótico

En la teoría del caos se describen comportamientos de sistemas dinámicos los cuales se ven afectados por sus condiciones iniciales [1]. En este caso tendremos unas condiciones iniciales llamadas $X_{i,0}$, donde i , es el número de llaves. Al tener 2 bytes por muestra, dividiremos esto en datos de 1 byte. Entonces el número de llaves serán 2.

El mapa caótico que se usó está definido en la ecuación 1.

$$X_{i,j} = bX_{i,j-1} + \left\lfloor \frac{X_{i,j-1}}{2^m} \right\rfloor + \epsilon \& \bigotimes_{k=1}^3 X_{k,j-1} \quad (1)$$

donde $\epsilon, b, m \in \mathbb{N}$ y $3 < m < 8$.

2.3.2. Algoritmo

El algoritmo creado para aplicar el mapa caótico de la ecuación 1 es el siguiente:

Input: *data, keys*

Output: *data*

```
1 map ← keys
2 for  $i = 1, n\_samples$  do
3   bytes ← data[i]
4   for  $j=1,2$  do
5     bytes[j] ← bytes[j]  $\otimes$  map[j]
6     map[j] ←  $f(map[j], m, \epsilon, b)$ 
```

3. Referencias

- [1] Edward Ott. *Chaos in dynamical systems*. Cambridge University Press, Cambridge, U.K. New York, 2002.