



Genera WAV

Es común que los archivos de audio sean capturados para ser reproducidos en un tiempo posterior, pero hay aplicaciones diversas en donde el sonido debe ser generado automáticamente, por ejemplo cuando se genera música MIDI o cuando se genera voz.

Este proyecto debe generar un programa en Java que dado un conjunto de parámetros, sea capaz de generar sonido con formato WAV, lo suficientemente estándar para poder ser reproducido en cualquier reproductor.

¿Cómo se hace?

Una señal de audio analógica es necesario digitalizarla para poder almacenarla en los medios digitales actuales. Un archivo de audio no es otra cosa que una colección de números que indican la amplitud de la señal en un instante cualquiera.

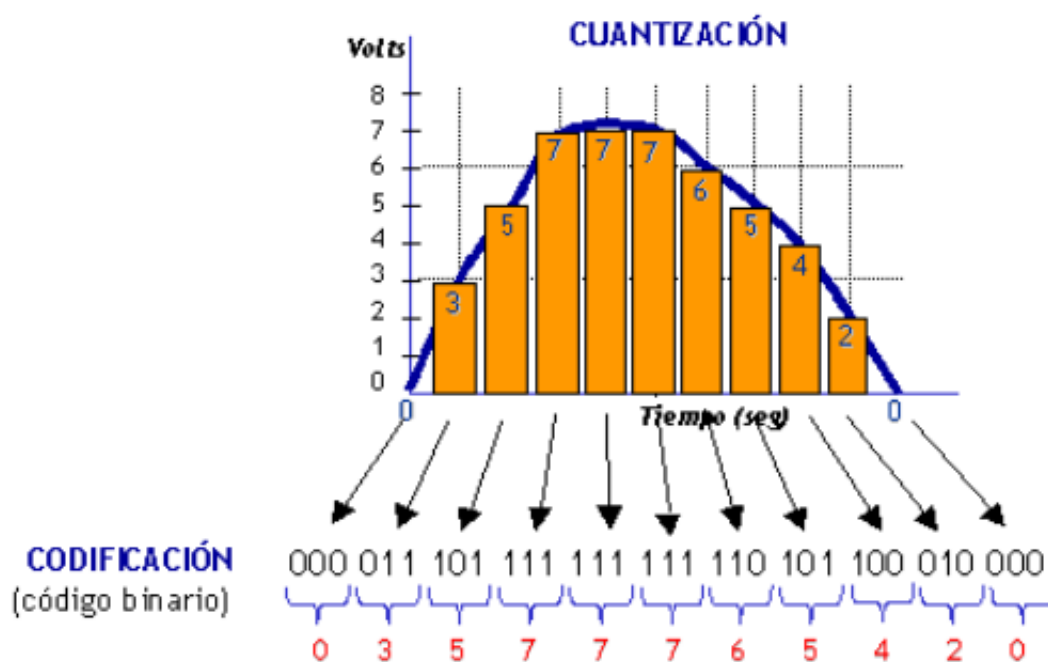


Figura 1: Señales discretas.

La señal analógica es continua con el tiempo, pero en una computadora, ni las señales, ni el tiempo son continuos, sino discretos. Cada cierto tiempo, se infiere sobre el valor de la señal analógica, lo cual recupera un cierto valor, llamado muestra. La velocidad con la que se recuperan muestras se llama frecuencia de muestreo, y entre más alto es su valor, mayor número de muestras hay que tomar.

Por otra parte, el número muestreado, se representa con un conjunto de bits (si la máxima amplitud es 8, basta con 3 bits). Cuando se almacena audio digital es necesario especificar: ¿Cuál es la velocidad de muestreo?, ¿Cuántos bits se usan para representar la amplitud?, ¿Cuántos bytes se necesitan para representar el valor de una muestra? Del mismo modo, en los formatos modernos, pueden almacenarse múltiples canales en el mismo archivo, por tanto, también se debe especificar cuántos canales tendrá la información. Las señales estéreo poseen dos canales.

El formato RIFF es utilizado para especificar diversos formatos de audio. Uno de ellos es el WAV. Un archivo WAV posee la estructura representada en la Figura 3).

El Cuadro 1 muestra la forma de como se almacenan los datos en el formato indicado.



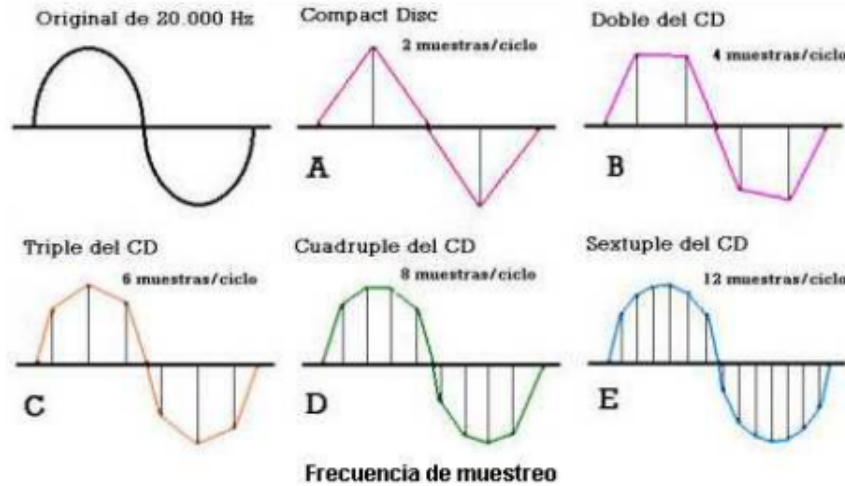
Proyecto 3

Universidad Nacional Autónoma de México

Programación Orientada a Objetos

Grupo 1

Fecha de entrega: viernes 22 de noviembre de 2024



La toma A, es muestreada a 40000 muestras por segundo

Figura 2: Frecuencias de muestreo.

Para este proyecto, se deberá generar una señal senoidal monoaural. En el ejemplo que se ha puesto como referencia, se usa el tono de la nota *La*.

En la Figura 4 se muestra la salida de la onda especificada con una frecuencia de la senoide almacenada de 440, una frecuencia de muestreo de 44,100, una amplitud de 32,000 a 16 bits de resolución.

Del mismo modo en la Figura 5 se muestra el contenido del archivo desde la cabecera y los contenidos que siguen después.

Especificaciones Técnicas

El código se invocará por consola en salida estándar.

La API de la clase GeneraWAV es:

```
public class GeneraWAV {
    . public void Escribe(String, int, int, int) {
    .     //Código del estudiante...
    . }
}
```

Los argumentos son:

- **String nombre:** nombre del archivo a crear.
- **int iTiempo:** segundos de duración del sonido a generar.
- **int iFrecuenciaMuestreo:** frecuencia de muestreo.
- **int iArmonico:** frecuencia de la señal a generar.

Se considera necesario realizar las validaciones usando una excepción de tipo `java.lang.IllegalArgumentException()`. La clase anterior debe ser invocada por la clase Proyecto3:

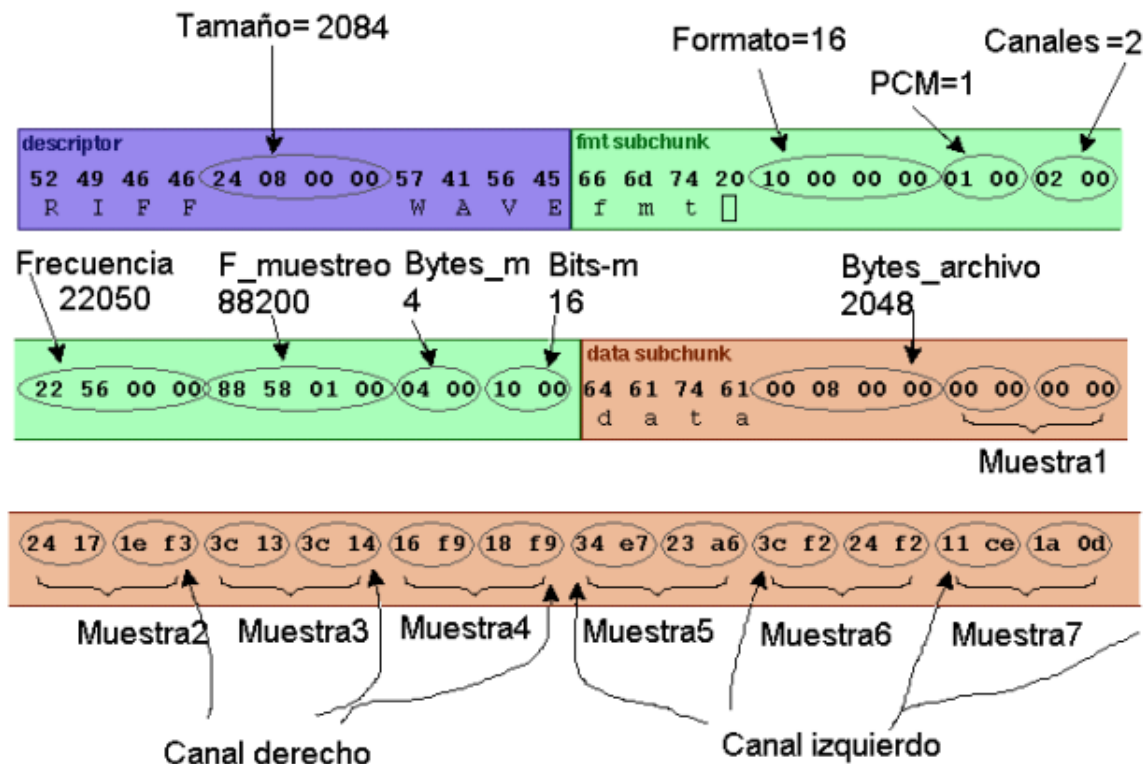


Figura 3: Formato RIFF

```
public class Proyecto3 {
    . public static void main(String[] args) {
    .     //Código del estudiante...
    . }
}
```

La aplicación va a recibir como entrada un archivo txt con los datos que se proveen para la construcción de la onda y de las muestras.

```
~$cat prueba.txt
Prueba.wav
22050
1
440
10
```

Donde:

- Prueba.wav es el nombre del archivo a generar.
- 22050 es la frecuencia de muestreo, expresada en Hertz.
- 1 es el número de canales (para este desarrollo, asuma que solo se desea el sonido en monoaural).



Proyecto 3

Universidad Nacional Autónoma de México

Programación Orientada a Objetos

Grupo 1

Fecha de entrega: viernes 22 de noviembre de 2024

Tipo	Variable	Contenido	Observaciones
char	Riff	'RIFF'	Constante (Big Endian)
long	Tamaño	Tamaño del archivo -8	Expresado en bytes (Little Endian)
char	Wave	'WAVEfmt'	Constante (Big Endian)
long	Formato	16	Constante. Expresa el número de bits (Little Endian)
short	PCM	1	PCM=1 (Little Endian)
short	Canales	Número de canales	Canales: mono(1), estéreo(2) (Little Endian)
int	Frecuencia	Frecuencia de muestreo	Expresada en Hz (Little Endian)
int	fMuestreo	Número de bytes por segundo	(Little Endian)
short	bytesM	Número de bytes por muestra	(Little Endian)
short	bitsM	Número de bits por muestra	(Little Endian)
char	Data	'data'	Constante (Big Endian)
int	bytesArchivo	Número de bytes que ocupan las muestras	(Little Endian)

Cuadro 1: Formato RIFF

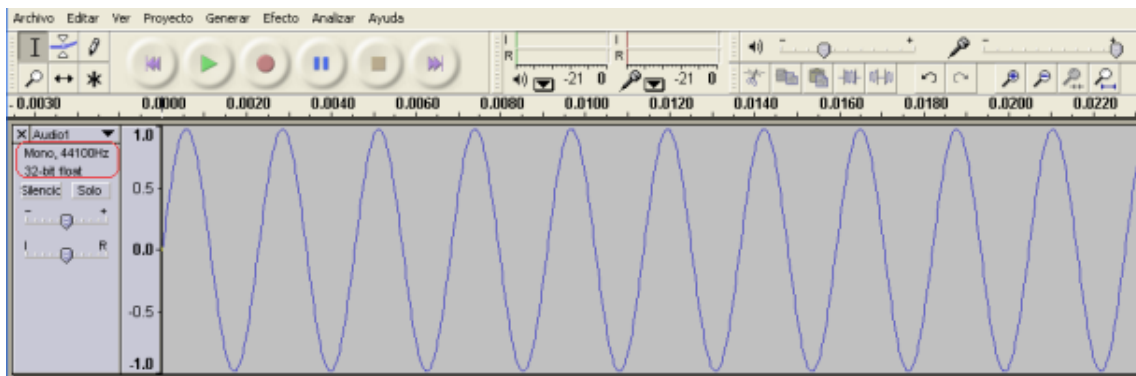


Figura 4: Salida de onda.

- 440 es el número de muestras.
- 10 es el tiempo que debe durar el sonido, expresado en segundos.

Cada argumento viene en una línea del archivo a leer. Se puede asumir que no habrá más argumentos en la línea y que el tipo de dato corresponde al indicado en cada uno de los datos aquí indicados.

```
~$java Proyecto3 prueba.txt
```

Las funciones de asignación de valores deben lanzar una excepción tipo *NullPointerException* si el cliente intenta asignar un valor nulo. Cuando los métodos de asignación reciban un argumento no permitido, se debe lanzar una excepción tipo *IllegalArgumentException*.

Se puede asumir que el nombre del archivo, cuando sea legal, llevará la extensión .wav. Se puede asumir que el archivo no intentará crearse en un medio de solo lectura. Se dotará de un .jar para verificar que el audio está correctamente formado.

Especificaciones de entrega

- La aplicación se sube a Classroom en su respectiva asignación de *proyecto* como: GeneraWAV.java, Proyecto3.java



Proyecto 3

Universidad Nacional Autónoma de México

Programación Orientada a Objetos

Grupo 1

Fecha de entrega: viernes 22 de noviembre de 2024

[illegible]

Figura 5: Contenido de Archivo

- El reporte se sube en archivo **PDF** con base en los lineamientos de entrega de reportes, se debe nombrar el archivo de la siguiente forma: NombreApellidoPaterno-Proyecto3.pdf
- Recordemos las ponderaciones y reglas expuestas en el Syllabus.

Frequently Asked Questions

- ¿La actividad es en equipo? No, es individual.
- ¿Puedo usar bibliotecas específicas del lenguaje? No, solo lo que sugiere en estas especificaciones y lo que se vió en clase durante este parcial.
- ¿Cuál es la ponderación en la evaluación? 60 % el código, 20 % el reporte y 20 % el cuestionario.