

PROYECTO 1

202208521 - Luis Rodrigo Morales Florián

1 Descripción del proyecto

Para la siguiente señal de audio, se genera la matriz de frecuencia siguiente:

TIEMPO (seg)	AMPLITUD (db)				
		1	2	3	4
	1	2	3	0	4
	2	0	0	6	3
	3	3	4	0	2
	4	1	0	1	5
	5	0	0	3	1

A continuación, se construye la matriz de patrones correspondiente a la señal de audio:

TIEMPO (seg)	AMPLITUD (db)				
		1	2	3	4
	1	1	1	0	1
	2	0	0	1	1
	3	1	1	0	1
	4	1	0	1	1
	5	0	0	1	1

En la matriz de patrones se puede observar que las filas 1 y 3 tienen los mismos patrones de frecuencia, así como también las filas 2 y 5. Entonces, habrá tres grupos considerando el tercer grupo formado simplemente por la fila 4. La cardinalidad de un grupo es definida por el número de tuplas incluidas en él.

La matriz reducida de frecuencias se obtiene sumando las tuplas en los grupos identificados en la matriz de patrones, esta matriz reducida será:

Project Description

For the following audio signal, the following frequency matrix is generated:

TIEMPO (seg)	AMPLITUD (db)				
		1	2	3	4
	1	2	3	0	4
	2	0	0	6	3
	3	3	4	0	2
	4	1	0	1	5
	5	0	0	3	1

Next, the pattern matrix corresponding to the audio signal is constructed:

TIEMPO (seg)	AMPLITUD (db)				
		1	2	3	4
	1	1	1	0	1
	2	0	0	1	1
	3	1	1	0	1
	4	1	0	1	1
	5	0	0	1	1

In the patterns matrix it can be seen that rows 1 and 3 have the same frequency patterns, as well as rows 2 and 5. So, there will be three groups considering the third group formed simply by row 4. The cardinality of a group is defined by the number of tuples included in it.

The reduced frequency matrix is obtained by adding the tuples in the groups identified in the pattern matrix, this reduced matrix will be:

		AMPLITUD (db)			
TIEMPO (seg)		1	2	3	4
	Grupo 1 (tiempo 1 y 3)	5	7	0	6
	Grupo 2 (tiempo 2 y 5)	0	0	9	4
	Grupo 3 (tiempo 4)	1	0	1	5

Usted ha sido contratado por la Facultad de Ingeniería para diseñar un programa que acepte “n” señales de audio (expresadas como matrices de frecuencia) y por cada una de estas señales de audio debe obtener las matrices reducidas con la misma matriz de patrones.

2 Gráficas

Se deberá utilizar la herramienta Graphviz para crear un grafo que muestre de manera gráfica las señales de audio ingresadas desde el archivo de entrada procesado y la matriz reducida producida a partir de la misma.

La figura 1 y 2 presentan un ejemplo de las gráficas a generar.

En la figura 1, “Prueba 1” será el nombre de la señal de audio, “t” y “A” serán las dimensiones (tiempo y amplitud) de la señal de audio y en la parte derecha se mostrarán las frecuencias que contiene la matriz de frecuencias de dicha señal.

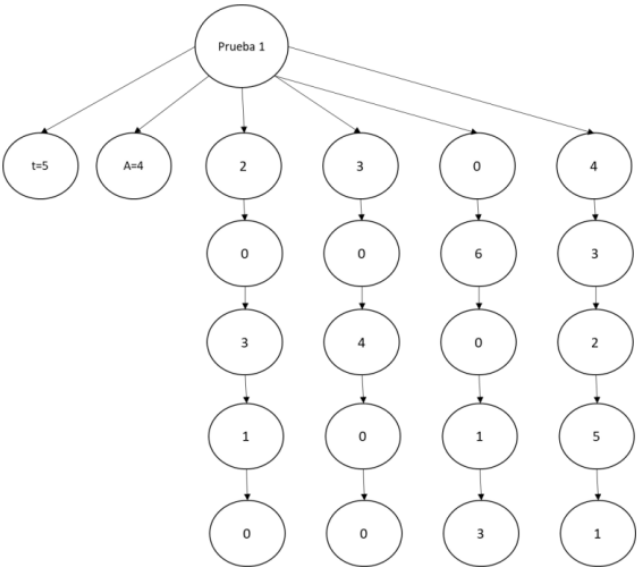


Figure 1: Onda original de la señal de audio

		AMPLITUD (db)			
TIEMPO (seg)		1	2	3	4
	Grupo 1 (tiempo 1 y 3)	5	7	0	6
	Grupo 2 (tiempo 2 y 5)	0	0	9	4
	Grupo 3 (tiempo 4)	1	0	1	5

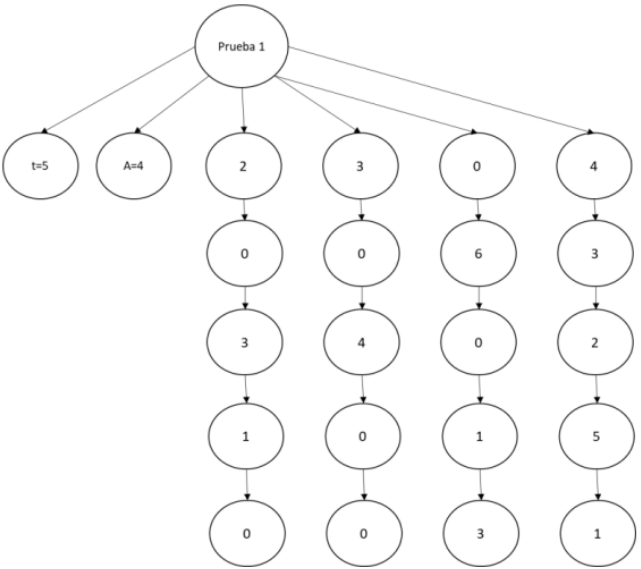
You have been hired by the Faculty of Engineering to design a program that accepts “n” audio signals (expressed as frequency matrices) and for each of these audio signals you must obtain the reduced matrices with the same pattern matrix.

Graphs

The Graphviz tool should be used to create a graph that graphically displays the input audio signals from the processed input file and the reduced matrix produced from it.

Figures 1 and 2 present an example of the graphs to be generated.

In figure 1, “Test 1” will be the name of the audio signal, “t” and “A” will be the dimensions (time and amplitude) of the audio signal, and on the right side, the frequencies contained in the frequency matrix of said signal will be displayed.



Original waveform of the audio signal

En la figura 2, se presenta la matriz reducida para la señal “Prueba 1”, “g” y “A” serán las dimensiones (grupos y amplitud) de la señal de audio y en la parte derecha se mostrará la matriz de frecuencias reducida para la señal de audio.

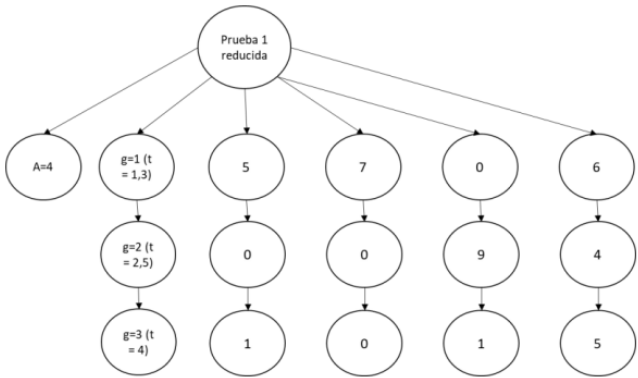


Figure 2: Onda reducida de la señal de audio

3 Diagrama de clases

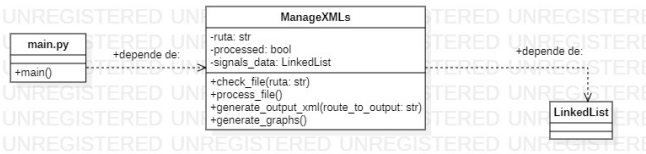
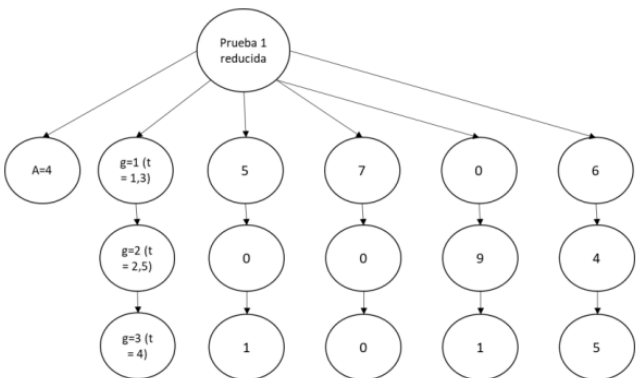


Figure 3: Diagrama de clases

A continuación se explicará cada clase:

In figure 2, the reduced matrix for the signal "Test 1" is presented, where "g" and "A" will represent the dimensions (groups and amplitude) of the audio signal, and on the right side, the reduced frequency matrix for the audio signal will be displayed.



Reduced waveform of the audio signal

Class Diagram

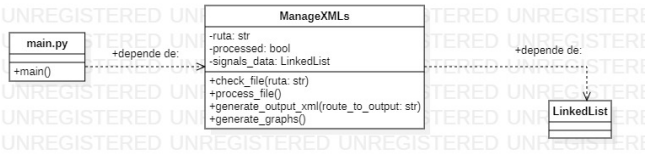


Diagrama de clases

Each class will be explained below:

3.1 main.py

Contiene el método main el cual es donde se inicializa toda la aplicación. Este método contiene el núcleo del programa el cual consiste en un menú de la siguiente forma:

1. **Cargar Archivo:** El usuario carga un archivo XML con la estructura indicada.
2. **Procesar Archivo:** El usuario selecciona un archivo y se procesa ese archivo seleccionado
3. **Escribir archivo salida:** El usuario selecciona un archivo (que ya esté procesado) y se crea un archivo XML con los datos procesados.
4. **Mostrar datos de estudiante:** El usuario selecciona un archivo (que ya esté procesado) y se crea un archivo XML con los datos procesados.
5. **Generar Gráfica:** El usuario selecciona un archivo (que ya esté procesado) y se le solicita que seleccione una señal del archivo. Se generan dos gráficas: La de los datos tal y como se ingresaron en la señal seleccionada del archivo ingresado de la opción 1, y la gráfica con los datos procesados

main.py

It contains the main method which is where the entire application is initialized. This method contains the core of the program which consists of a menu as follows:

1. ***Load file:*** *The user uploads an XML file with the indicated structure.*
2. ***Process file:*** *The user selects a file and the selected file is processed.*
3. ***Write output file:*** *The user selects a file (which is already processed) and an XML file is created with the processed data.*
4. ***Show student data:*** *The user selects a file (which is already processed) and an XML file is created with the processed data.*
5. ***Generate Graphs*** *The user selects a file (which is already processed) and is prompted to select a signal from the file. Two graphs are generated: The one with the data as they were entered in the selected signal from the file entered in option 1, and the graph with the processed data.*

3.2 ManageXMLs

3.2.1 Atributos

Estos son los atributos que sus valores son establecidos en el constructor

1. **ruta: str:** Es la ruta del archivo XML de donde se obtendrá los datos iniciales
2. **processed: bool:** El estado de los datos iniciales ingresados. Si es 'True', los datos han sido procesados. Por defecto, empieza con un valor 'False', lo que indica que inicialmente no están procesados y por ende, se deben de procesar.
3. **signals_data: LinkedList:** Cuando ingresamos un archivo XML, se sabe que éste puede traer más de una señal. Por tanto este es un contenedor tipo LinkedList que va almacenando cada una de estas señales

3.2.2 Métodos

Estos son los atributos que sus valores son establecidos en el constructor

1. **check_file(ruta: str):** Chequea si la ruta ingresada existe
2. **process_file():** Procesa el archivo previamente ingresado en el atributo 'ruta', es decir, 'ruta' no puede ser 'None'.
3. **generate_output_xml(route_to_output: str):** Genera un XML del archivo procesado previamente ingresado en el atributo 'ruta', es decir, 'ruta' no puede ser 'None' y el atributo 'processed' debe de ser 'True'
4. **generate_graphs():** Genera un gráficas de una señal en específico del archivo procesado previamente ingresado en el atributo 'ruta', es decir, 'ruta' no puede ser 'None', el atributo 'processed' debe de ser 'True' y debe de existir por lo menos una señal dentro del atributo 'signals_data' ya que se le va a requerir en el menú que elija una.

3.3 LinkedList

Es una clase que maneja LinkedList y todos sus respectivos métodos

ManageXMLs

Attributes

These are the attributes that their values are set in the constructor.

1. **ruta: str:** It is the path of the XML file from where the initial data will be obtained.
2. **processed: bool:** The status of the initial data entered. If 'True', the data has been processed. By default, it starts with a value of 'False', which indicates that they are not initially processed and therefore should be processed.
3. **signals_data: LinkedList:** When we input an XML file, it is known that it can bring more than one signal. Therefore this is a LinkedList type container that stores each of these signals

Methods

These are the attributes that their values are set in the constructor.

1. **check_file(ruta: str):** Check if the entered path exists.
2. **process_file():** Process the previously entered file in the 'ruta' attribute, that is, 'ruta' cannot be 'None'.
3. **generate_output_xml(route_to_output: str):** Generates an XML of the previously processed file entered in the 'path' attribute, ie 'path' cannot be 'None' and the 'processed' attribute must be 'True'.
4. **generate_graphs():** It generates a graph of a specific signal of the previously processed file entered in the 'path' attribute, that is, 'path' cannot be 'None', the 'processed' attribute must be 'True' and there must be at least one signal within the 'signals_data' attribute since it will be required in the menu to choose one.

LinkedList

It is a class that handles LinkedList and all its respective methods.