[[1]](#footnote-1)

Tarea #5

Francisco Alberto, Navarro Orozco, 202004752

*Escuela de Mecánica Eléctrica, Facultad de Ingeniería, Universidad de San Carlos de Guatemala*

*Para la quinta tarea se realizó un código que sea capaz de grabar la voz, reproducir el audio y realizar las gráficas*

# Código utilizado

if(exist('OCTAVE\_VERSION','builtin')~=0)

pkg load signal;

end

%MENU

opcion = 0;

while opcion ~=5

disp('Seleccione una opcion')

disp('1.Grabar')

disp('2.Reproducir')

disp('3.Graficar')

disp('4.Graficar densidad')

disp('5. Salir')

opcion = input('Ingrese su elección:');

switch opcion

case 1

try

duracion = input('Ingrese la duración de la grabación en segundos:');

disp('Comenzando la grabación');

recObj = audiorecorder;

recordblocking(recObj, duracion);

disp('Grabacion finalizada');

data= getaudiodata(recObj);

audiowrite('audio.wav', data, recObj.SampleRate);

disp('Archivo de audio grabado correctamente');

catch

disp('Error al grabar audio');

end\_try\_catch

case 2

try

[data, fs] = audioread('audio.wav');

sound(data, fs);

catch

disp('Error al reproducir el audio');

end\_try\_catch

case 3

try

[data, fs]=audioread('audio.wav');

tiempo = linspace(0, length(data)/fs, length(data));

plot(tiempo, data);

xlabel('Tiempo(s)');

ylabel('Amplitud');

title('Audio');

catch

disp('Error al graficar el audio');

end\_try\_catch

case 4

try

disp('Graficando espectro de frecuencia');

[audio, Fs] = audioread('audio.wav');

N = length(audio);

f = linspace(0, Fs/2, N/2+1);

ventana = hann(N);

Sxx= pwelch(audio, ventana, 0, N, Fs);

plot(f, 10\*log10(Sxx(1:N/2+1)));

xlabel('Frecuencia (Hz)');

ylabel('Densidad espectral de potencia(dB/Hz)');

title('Espectro de freuencia de la señal grabada');

catch

disp('Error al graficar audio');

end\_try\_catch

case 5

disp('Saliendo del programa');

break

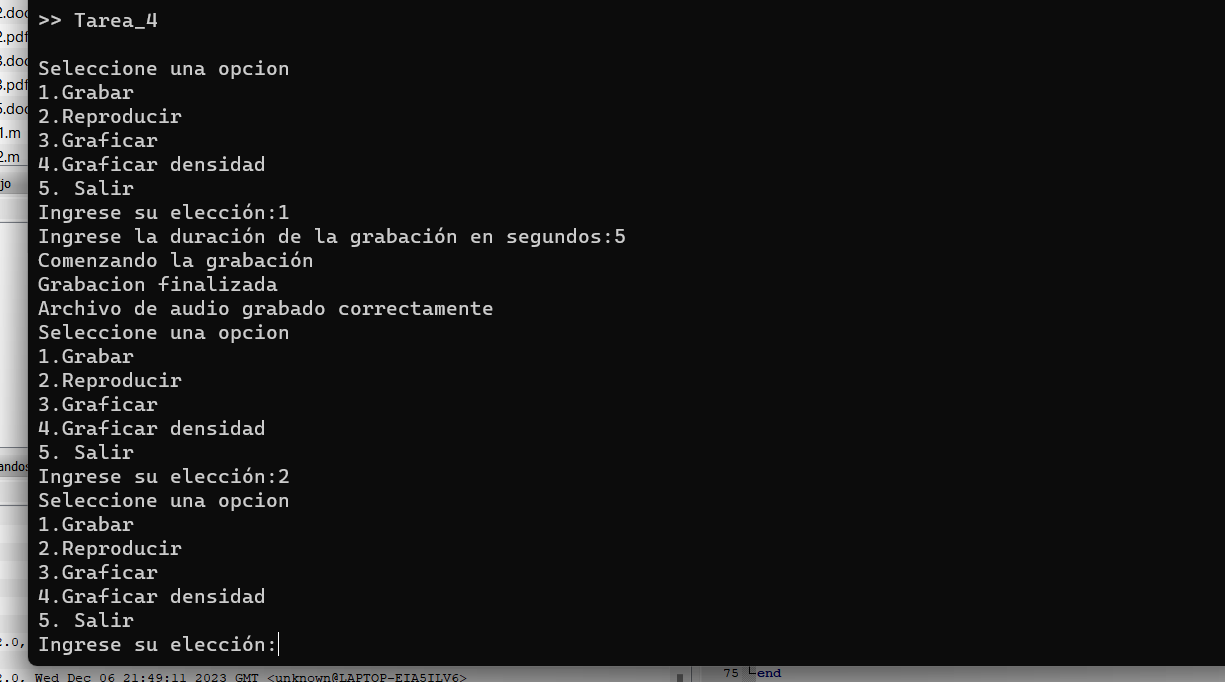
otherwise

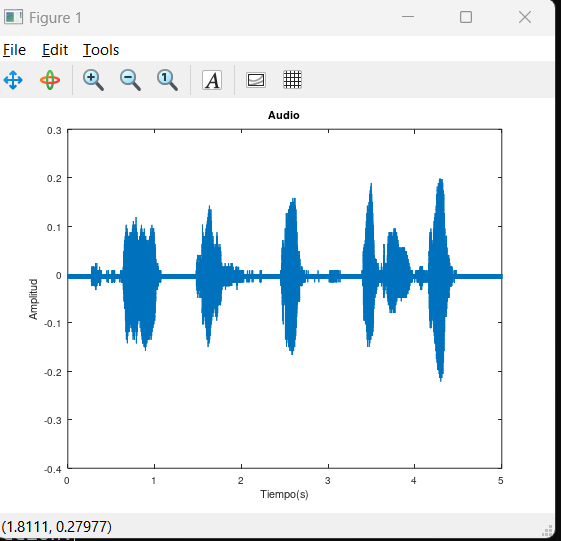
disp('Opción inválida');

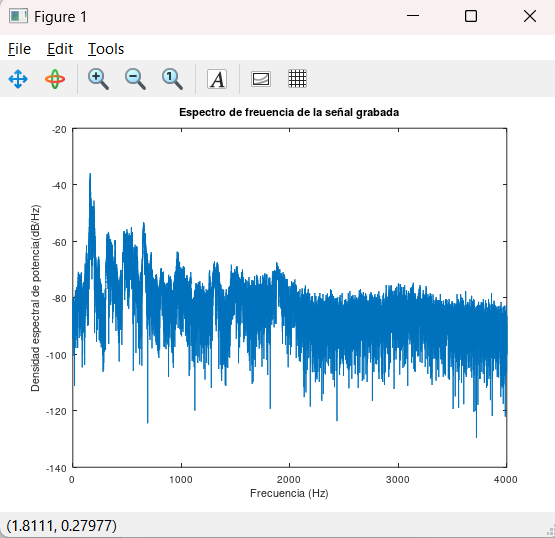
end

end

# Resultados







1. [↑](#footnote-ref-1)