EE 242 Lab 0 – Texts, Code, Plot and Play

In this lab exercise, basic python tools that will be required throughout this course to perform the various lab assignments.

Form Teams

All in-lab sessions will be conducted as a team of 2-4 students. We prefer teams of 3 over other combinations.

Write down the names and number of your team assigned by the TA of the form "XXYY" where XX is your lab session code (AA, AB, AD, AE, AF) and YY is your team number.

Team AA4: Anna Petrbokova, Leonard Paya, Henry Adams, Grace Hwang

Playing with strings

Lab 0.1 - Palindromes

Write the python code for reading input from user and write a function to check whether the string is a palindrome using the following algorithm

- read input in a string a
- reverse the string b
- write a for loop to check from 1 → N if a[i]=b[i]
- if all characters match return True else return False

In [1]:

```
#Python code goes here
a=input("Input A:")

# Function that checks if a string is a palindrome
def isPalindrome(input_string):
    reverseString = input_string[::-1]# store input_string to reverseString in reverse order fro
    count = 0
    for char in input_string:# loops over every character in input_string
        if input_string[count] != reverseString[count]:
            return False#return false once the character of the same index in input_string and re

        else:
            count += 1
    return True#returns True if it doesn't return False and finishes looping through all characte

b=isPalindrome("Mahmood Hameed");
print("Output B:"+str(b)) #Should be False
c=isPalindrome("Signals, Systems and Data I:I ataD dna smetsyS ,slangiS")
print("Output C:"+str(c)) #Should be True
d=isPalindrome(a)
print("Output D:"+str(d))
```

```
Output B:False
Output C:True
Output D:False
```

Questions to be answered in the report:

1. Is there a way to make this algorithm more efficient in time required and storage space?

While this algorithm will always take O(n) time to run, one can change this algorithim to constant storage space by replacing the flipped string with two pointers, one at the start and one at the end of the string, then moving the pointers closer together and checking if their respective characters are equal

2. Are there cases where the above algorithm/code will fail?

When considering if upper and lower case is the same or if puncuation should be removed when looking at string

Lab 0.2 - Teams

Write the python code for reading input from user and write a function to replace all vowels using the following encoding scheme:

- Aa ⟶ @
- Ee ⟶ 3
- Ii ⟶ 1
- Oo ⟶ 0
- Uu ⟶ ^

Show the TA example with all of your names concatenated in the string.

In [ ]:

```python
#python code goes here
a=input("Input A:")
def encode(input_string):
    #your code goes here
    vowel_map = {'a': '@', 'e': '3', 'i': '1', 'o': '0', 'u': '^',
                 'A': '@', 'E': '3', 'I': '1', 'O': '0', 'U': '^'}

    output_string=''.join([c if c not in vowel_map else vowel_map[c] for c in input_string])
    return output_string

b=encode("Mahmood Hameed");
print("Output B:" + b) #Should be M@hm00d H@m33d
c=encode("Signals, Systems and Data I: This is my first lab session")
print("Output C:" + c) #Should be S1gn@ls, Syst3ms @nd D@t@ 1: Th1s 1s my f1rst l@b s3ss10n
d=encode(a)
print("Output D:" + d)
```

```
Output B:M@hm00d H@m33d
Output C:S1gn@ls, Syst3ms @nd D@t@ I: Th1s 1s my f1rst l@b s3ss10n
Output D:Ohhhhhhhh I'm h@lf w@y th3r3, OHHHHH0h L1v1n 0n @ pr@y3r
```

Questions to be answered in the report:

1. What is the time complexity of the code if the length of the string is N?

The time complexity is O(N) since needs to check every character to replace it with the desired alternative

2. How much storage does your function require if the length of the string is N? (1 char is 1 byte)

The storage required is N bytes plus the bytes the map requires. If the length of a string is N then there are N characters so N characters times 1 byte is N bytes.

Playing with Signals

Lab 0.3 - Reading, playing and writing audio files

Use scipy package to read the provided baby.wav file. **Refer to documentation (https://docs.scipy.org/doc/scipy/reference/io.html)**

Perform these 4 actions :

- Load the wave file
- Plot the wave file in time-domain **Refer to this demo for matplotlib (https://matplotlib.org/stable/plot_types/basic/plot.html#sphx-glr-plot-types-basic-plot-py)**
- Write a new wave file called "babyhigh.wav" with a sample rate of 96000
- Write a new wave file called "babylow.wav" with a sample rate of 24000

Use IPython package to play the provided baby.wav file as well as the other two "babyhigh.wav" and "babylow.wav" files. **Refer to documentation (https://ipython.readthedocs.io/en/stable/api/generated/IPython.display.html)**

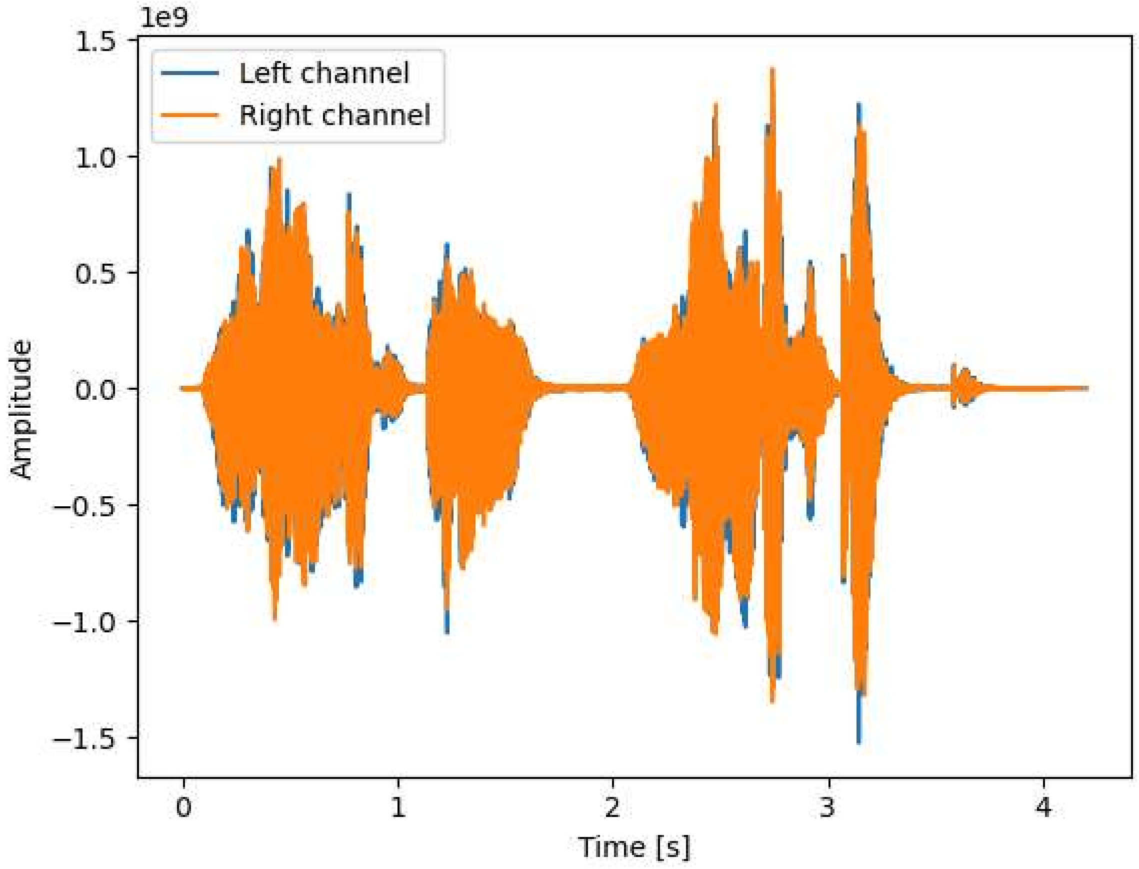
In [3]:

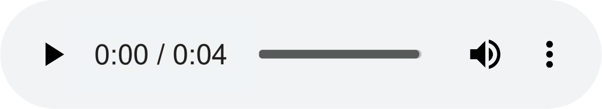
```
!python install --user IPython
!pip install scipy
!pip install matplotlib
import numpy as np
import matplotlib.pyplot as plt
import scipy.io.wavfile as wavfile
#Python Code goes here
#Install the IPython package using !python install --user IPython
#1. Read the baby.wav file.
samplerate, data = wavfile.read('baby.wav')

#2. Plot the laugh sound (What are the two arrays that you see?)
length = data.shape[0] / samplerate

time = np.linspace(0., length, data.shape[0])
plt.plot(time, data[:, 0], label="Left channel")
plt.plot(time, data[:, 1], label="Right channel")
plt.legend()
plt.xlabel("Time [s]")
plt.ylabel("Amplitude")
plt.show()

#3. Store the data with sample rate of 96000
wavfile.write('babyhigh.wav', 96000, data)

#4. Store the data with sample rate of 24000
wavfile.write('babylow.wav', 24000, data)

#5. Play all the files. Notice how the sound changes?
from IPython.display import Audio
display(Audio('baby.wav'))
display(Audio('babyhigh.wav'))
display(Audio('babylow.wav'))
```

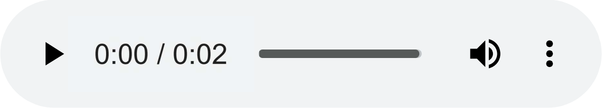
```
python: can't open file 'c:\\Users\\Leona\\OneDrive - UW\\2024-2025 Sophomore\\Winter 2025\\EE 24
2\\ee-242\\EE 242 Lab 0\\install': [Errno 2] No such file or directory
```

```
Requirement already satisfied: scipy in c:\users\leona\anaconda3\lib\site-packages (1.13.1)
Requirement already satisfied: numpy<2.3,>=1.22.4 in c:\users\leona\anaconda3\lib\site-packages
(from scipy) (1.26.4)
Requirement already satisfied: matplotlib in c:\users\leona\anaconda3\lib\site-packages (3.9.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\leona\anaconda3\lib\site-packages (fr
om matplotlib) (1.2.0)
Requirement already satisfied: cycler>=0.10 in c:\users\leona\anaconda3\lib\site-packages (from m
atplotlib) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\leona\anaconda3\lib\site-packages (f
rom matplotlib) (4.51.0)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\leona\anaconda3\lib\site-packages (f
rom matplotlib) (1.4.4)
Requirement already satisfied: numpy>=1.23 in c:\users\leona\anaconda3\lib\site-packages (from ma
tplotlib) (1.26.4)
Requirement already satisfied: packaging>=20.0 in c:\users\leona\anaconda3\lib\site-packages (fro
m matplotlib) (24.1)
Requirement already satisfied: pillow>=8 in c:\users\leona\anaconda3\lib\site-packages (from matp
lotlib) (10.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\leona\anaconda3\lib\site-packages (fr
om matplotlib) (3.1.2)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\leona\anaconda3\lib\site-packages
(from matplotlib) (2.9.0.post0)
Requirement already satisfied: six>=1.5 in c:\users\leona\anaconda3\lib\site-packages (from pytho
n-dateutil>=2.7->matplotlib) (1.16.0)
```

```
C:\Users\Leona\AppData\Local\Temp\ipykernel_43940\1834770681.py:10: WavFileWarning: Chunk (non-da
ta) not understood, skipping it.
  samplerate, data = wavfile.read('baby.wav')
```







Questions to be answered in this report:

1. Store the wavefile with a lower sampling rate and try to play it. Does it sound higher pitch or lower pitch? (Keep the IPython display open to show TA)

The lower sampling rate sounds lower pitch

2. Store the wavefile with a higher sampling rate and try to play it. Does it sound higher pitch or lower pitch? (Keep the IPython display open to show TA)

The higher sampling rate gas higher pitch