San Jose State University

Department of Electrical Engineering

EE104, Fall 2021, Pham

Laboratory Assignment #7

# Objectives

This lab introduces you the art of integration a few tools that you are already familiar to provide a business solution that you can use in the real life application using FFT.

# Grading

Refer to the section **Python Programming** for grading criteria.

# Bibliography

Refer to lecture notes for sample programs.

# Requirements

## 1 - FFT/IFFT Audio Signal Processing – Noise Cancelling Application

Using your own audio recording device such as your smart phone or computer to record and generate a WAV file that has at least 3 tones with different frequencies. You can also leverage the following websites for tone generation or write your own Python code to generate the tones:

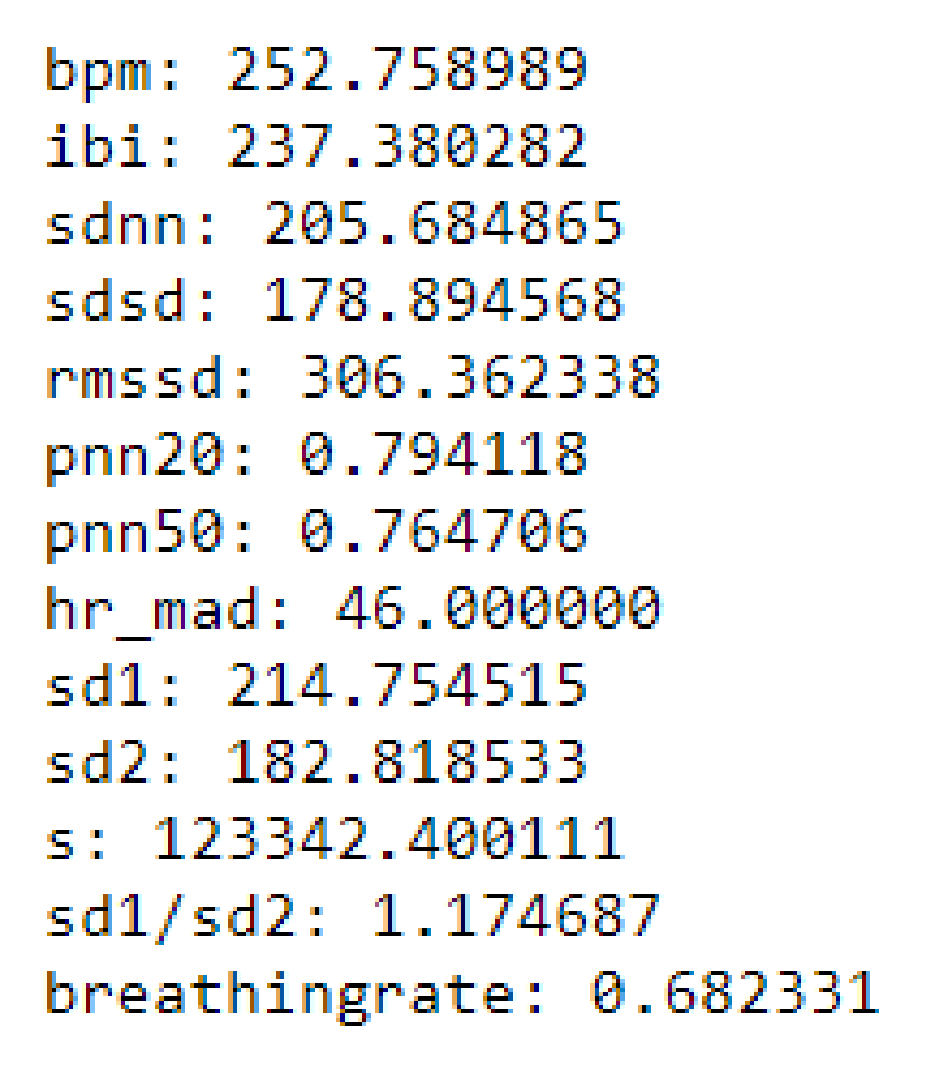
* <https://www.audiocheck.net/audiofrequencysignalgenerator_index.php>
* <https://www.wavtones.com/functiongenerator.php>

Write a Python program to show the signals in the frequency domain. Filter out one or more frequencies as you wish and save the file back in the CSV format, then convert the same file to WAV to prove that you successfully removed the “unwanted” sound(s).

## 2 - Heart Rate Analysis – Time Domain Measurements – Biotechnology

Download a WAV file from the heartbeat sound bank website <https://www.kaggle.com/kinguistics/heartbeat-sounds> by selecting a wave file that shows at the minimum 30 beats in the preview waveform. Using WAV to CSV conversion tool and write a Python program to plot the heart rate signal and find the time-domain measurements for that WAV file. See a sample below.

Note: CSV data has to be normalized to 0.xxx format. For example, if your converted data is 95, then normalize by dividing the number by 1000 to get 0.095.



## 3 - Heart Rate Diagnostic Analysis – Biotechnology

### Background:

This website <http://www.peterjbentley.com/heartchallenge/> has 176 files in WAV format in DatasetA, representing 5 major groups as follows:

**Dataset A,**containing 176 files in WAV format, organized as:

|  |  |  |  |
| --- | --- | --- | --- |
| Atraining\_normal.zip | 14Mb | 31 files | [download](http://www.peterjbentley.com/heartchallenge/wav/Atraining_normal.zip) |
| Atraining\_murmur.zip | 17.3Mb | 34 files | [download](http://www.peterjbentley.com/heartchallenge/wav/Atraining_murmur.zip) |
| Atraining\_extrahs.zip | 6.9Mb | 19 files | [download](http://www.peterjbentley.com/heartchallenge/wav/Atraining_extrahs.zip) |
| Atraining\_artifact.zip | 22.5Mb | 40 files | [download](http://www.peterjbentley.com/heartchallenge/wav/Atraining_artifact.zip) |
| Aunlabelledtest.zip | 24.6Mb | 52 files | [download](http://www.peterjbentley.com/heartchallenge/wav/Aunlabelledtest.zip) |

### Requirements:

1. Perform needed research in the heart rate analysis domain to understand the key parameters to determine a certain heart rate syndrome such as heart murmur, etc. Provide a write-up summarizing those parameters. Cite the reliable medical resources where you obtain your information.
2. Design and implement a model using Python programming language to analyze 1 sample WAV file from each group from the same Dataset A from the website <http://www.peterjbentley.com/heartchallenge/> to print out a suggestion to the medical staff that your Python program detects a heart rate anomaly and print out all parameters that you can have to support your analysis. Hint: The file names already suggest the heart symptoms.

# Python Programming

#### Lab Submission

|  |  |  |
| --- | --- | --- |
| **Program or Requirement** | **Use Case** | **Earned Score / Max Score** |
| README file and Documentation on Github | This is a brief user guide so that the user can install the proper python packages and knows how to execute your program. The README file can contain sample screenshots with explanation. | \_\_\_\_\_ / 10 |
| FFT/IFFT Audio Signal Processing – Noise Cancelling Application with minimum one code execution on Google Lab | Acoustic noise-cancelling headsets | \_\_\_\_\_ / 30 |
| Heart Rate Analysis – Time Domain Measurements – Biotechnology with minimum one code execution on Google Lab | Hospital clinical vital measurement instrument | \_\_\_\_\_ / 30 |
| Heart Rate Diagnostic Analysis – Biotechnology with minimum one code execution on Google Lab | Hospital clinical diagnostic instruments | \_\_\_\_\_ / 30 |
|  | **TOTAL** | **100%** |

That’s all for this lab. Hopefully you found it useful and increase your interest in the Python world! See you in the next lab.

# Laboratory Hand-In Requirements

Once you have completed a working design, prepare for the submission process. You are required to demonstrate a working design. You are also required to submit an archive of your project in the form of a ZIP file. Use 7-Zip option to create the ZIP file. Name the archive lab#\_yourlastname\_yourfirstname.zip. Refer to Lab 1 for detail instructions.

You will submit your zip file to the instructor through Canvas by the due date and time. You will also submit the same to Github so that you can run the Google Colab directly from Github. If the class will be on campus, then you will expect to demonstrate in the classroom. If we ever have to go back to an online mode, turn in your archive to Canvas along with a narrated video capturing the screen of your computer running your program demonstration. If your program is not completely functional by the due date, you should demonstrate and turn in what you have accomplished to receive partial credit. See the syllabus for the late penalty guideline