Winnie Leung
EE104 Sec 01
Professor Pham
April 22, 2022

LAB 7 README

Abstract

****

The purpose of this lab is to introduce users to the art of integration for fast fourier transform (FFT) applied in real world applications for business solutions.

Objective

****

Fulfill all objectives listed below

Program or Requirement	Use Case	Earned Score / Max Score
Demonstration Video	You must submit a demonstration video or your score for this lab will be zero	
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. You will practice how to use GitHub.	_____ / 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
Game Development – Red Alert	Entertaining Industry, Education	_____ / 30
	TOTAL	**100%**

Requirements

****

FFT/IFFT Audio Signal Processing – Noise Canceling Application

```
import numpy as np
from scipy import fftpack, signal
from matplotlib import pyplot as plt
import pandas as pd
from scipy.io import wavfile
```

Heart Rate Analysis – Time Domain Measurements – Biotechnology

- Pip install heartpy

```
import matplotlib.pyplot as plt
import heartpy as hp
```

Pygame- Red Alert

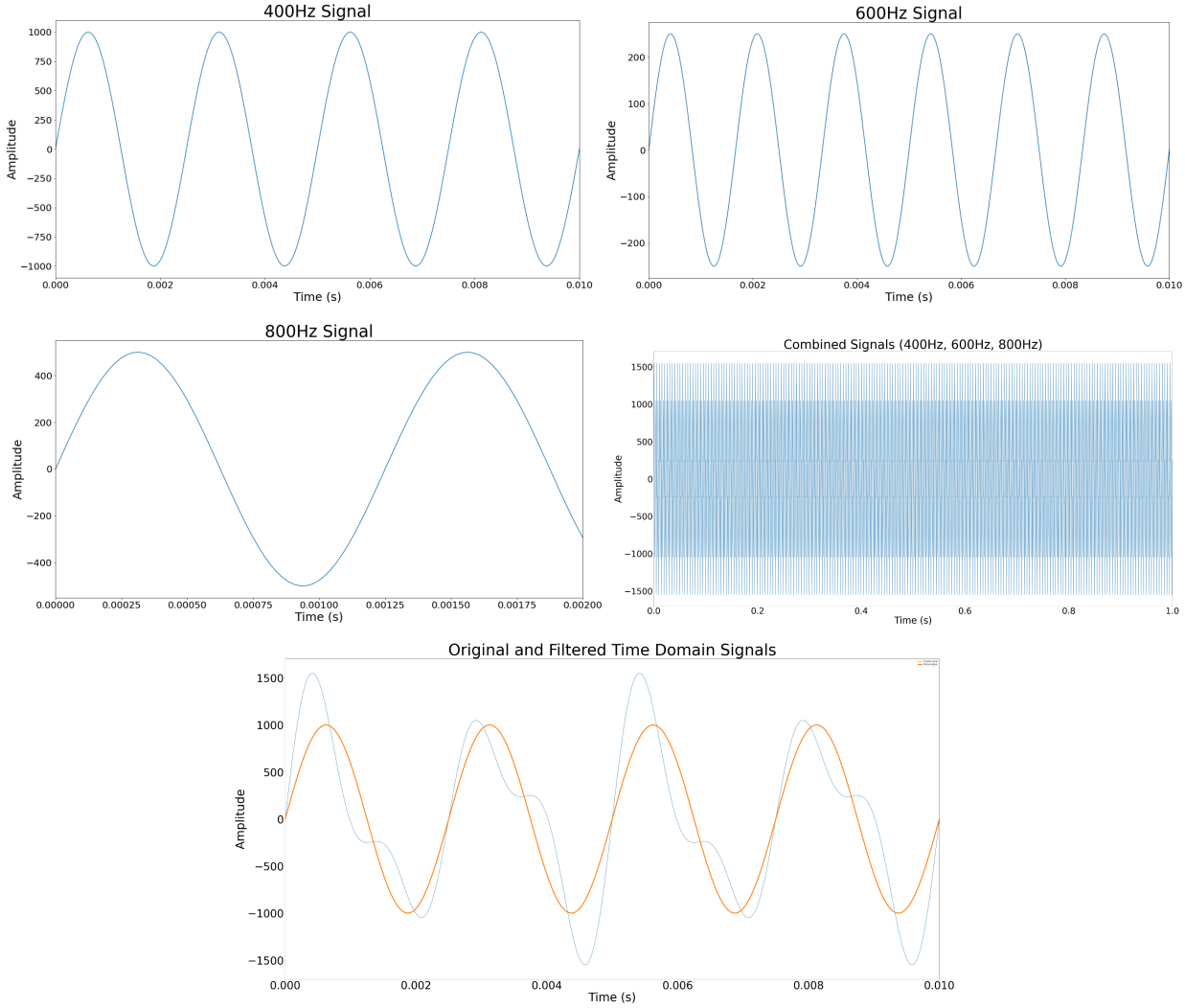
-

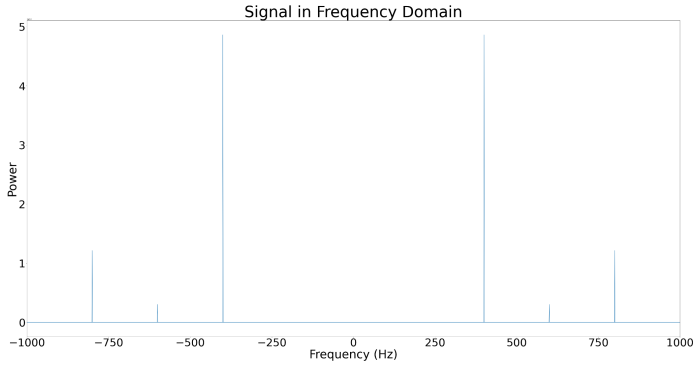
```
import pgzrun
import pygame
import pgzero
import random
from pgzero.builtins import Actor
from random import randint
```

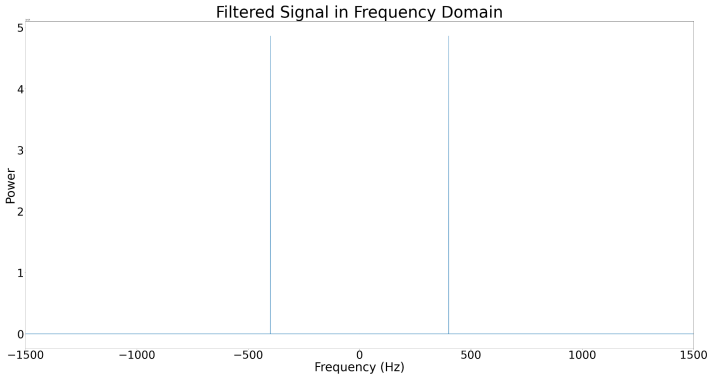
Instructions

****

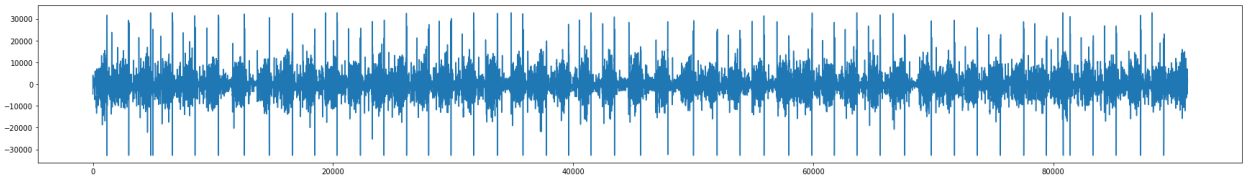
1. Download all the required packages prior to running the code
2. FFT/IFFT Audio Signal Processing – Noise Canceling Application
    a. Open the FFT.py program and run the code
    b. The output plots should display the following

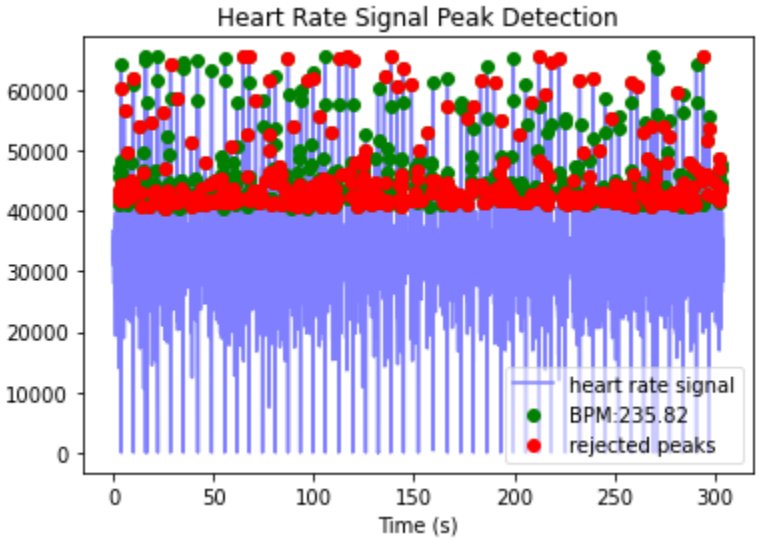






3. Heart Rate Analysis
    a. Download and run the program HEARTBEAT.py
    b. The output of the program should look like the following

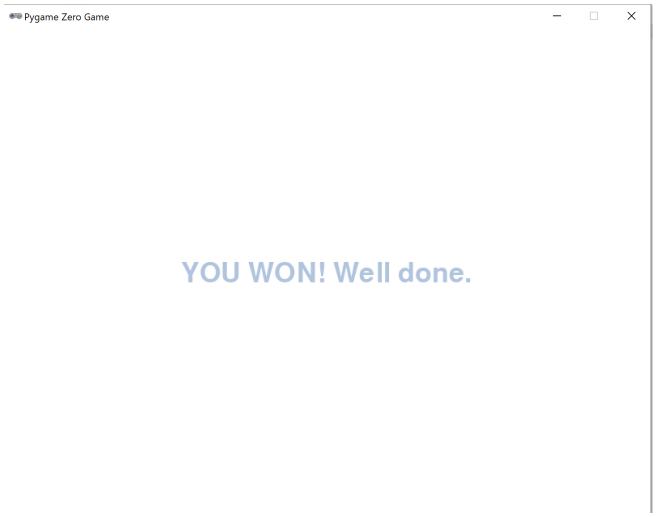





```
bpm: 235.824206
ibi: 254.426808
sdnn: 155.428834
sdsd: 143.438385
rmssd: 253.327377
pnn20: 0.888889
pnn50: 0.888889
hr_mad: 86.666667
sd1: 177.920510
sd2: 138.032906
s: 77154.012696
sd1/sd2: 1.288972
breathingrate: 0.233333
```

4. Pygame- red alert
    a. Download and run the program red.py
    b. The final output should look like the following

Level 5



If red snowflake was selected after 6 levels

GAME OVER! Try again.

If non-red snowflakes were clicked on