CPE 381: Fundamentals of Signals and Systems for Computer Engineers.

Project

Programming Assignment Phase 1.

Submitted by: Dan Otieno.

Due Date: 03/10/23

Introduction:

The purpose of this project is to work with and understand sample processing of signals. We use MATLAB and C/C++ to analyze a sample, which is provided as a voice recording tentatively set at 14 seconds in length. The final 7 seconds of the audio signal is modified in MATLAB and we use C++ to analyze the modified file and output an analysis summary in a text file.

Results & Observation:

Assignment:

Description:

The assignment instructions are to record an audio (WAV) file which is 14 seconds long at a frequency f=11025Hz, and then read that WAV file in MATLAB to replace the signal from time t=7sec as follows:

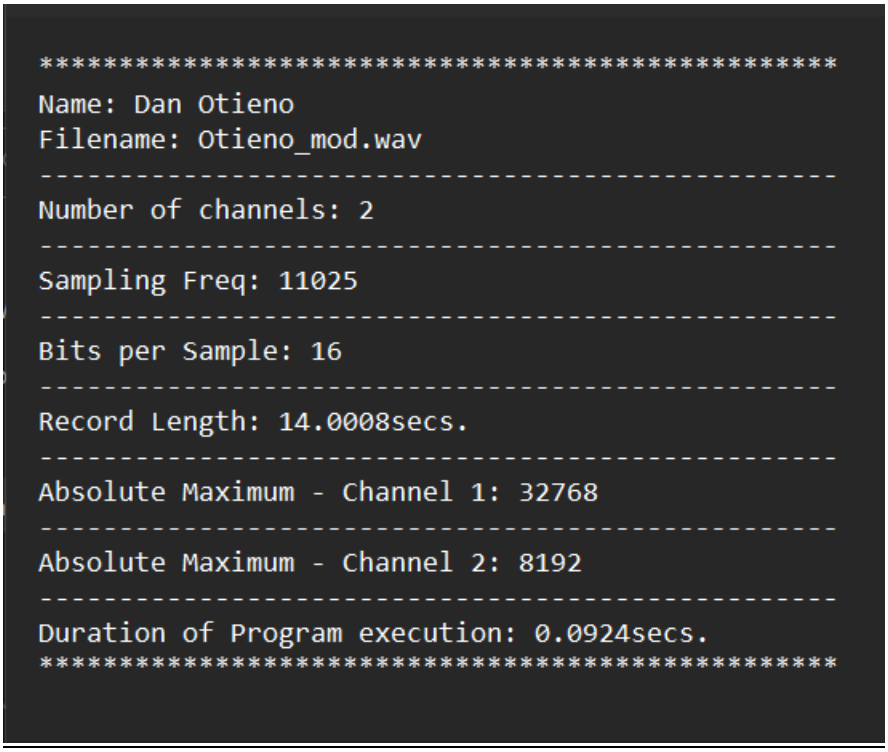
- Left channel will have sine wave at f=1,810 Hz, amplitude equal to 0.3, and initial phase of π/4.
- Right channel will have sine wave at f=1,990 Hz, amplitude equal to 0.25, and initial phase of 3π/4.

The next step instructs to write a C/C++ program that reads the modified file (after user input in command line), opens, processes, writes and outputs a summary in a text file.

Program Outputs:

- **Assignment 1 outputs:**

```
dpo0002prac@DESKTOP-14ODFI9:/mnt/c/Users/d_oti/Desktop/CPE_CLASSES/CPE_381/Project$ ./phs1
Enter input filename: Otieno_mod.wav
Reading from file path, please wait....
Size of File: 617444
-------------------------------------
Size of Header(bytes): 1
-------------------------------------
Bits per Sample: 16
-------------------------------------
Bytes per sample: 2
-------------------------------------
Samples: 308718
-------------------------------------
Enter summary filename: Otieno_D_info.txt
Printing summary, please wait....
Processing complete...
dpo0002prac@DESKTOP-14ODFI9:/mnt/c/Users/d_oti/Desktop/CPE_CLASSES/CPE_381/Project$
```



More Details:

The control fields for the WAV file are found in the Struct function of the C++ file. A WAV file comprises a header and data. This program cannot work in real time because the signal processing would require that the analyzed input and generated output samples can be processed continuously within the time taken to input and output the same set of samples, without significant delays. However, from my text summary, the program execution time is just under 0.1secs, which would make it a relatively quick run time.