CSCI366 Multimedia Computing

Assignment One (15%)

Due Date: (please check with Dr Loo)

Tasks

You are required to develop a program to mix two audio signals and then play the mixed audio Let $f_1(t)$ and $f_2(t)$ be the two audio signals. The mixed audio $g(t)$ is

$$g(t) = \alpha f_1(t) + (1-\alpha) f_2(t)$$

where $0 \leq \alpha \leq 1.0$ is the mixing parameter. Notice that the length of $f_1(t)$ and $f_2(t)$ may be different. If their lengths are T_1 and T_2 respectively, The length of $g(t)$ signal, T, should be $max(T_1, T_2)$. The shorter signal should be repeated in order to generate $g(t)$. In addition, the format of $f_1(t)$ and $f_2(t)$ may also be different. Your program should be able to mix any 16 bits mono and stereo signals as long as the two signals have the sample sampling frequency. When one of the signals $f_1(t)$ and $f_2(t)$ is stereo, the mixed signal $g(t)$ should be stereo as well. When both signals are mono, then $g(t)$ is also mono. In the cases that $f_1(t)$ and $f_2(t)$ do not have the same sampling frequency or one of them is not 16 bits, the program should report that the program does not support the format of the input signals and display the formats and other information on both $f_1(t)$ and $f_2(t)$

Requirements

1. The program should be named as "**mixplay**" and shall read α value from the command lines and take two WAV audio file as the input audio $f_1(t)$ and $f_2(t)$. For instance, the command line may look like this

   `$mixplay` α *`f1.wav f2.wav`*

2. The program should just display the usage of the program and exists when it is invoked as follows

   `$ mixplay –help`

3. The program should support 16 bit mono/stereo WAV files of same sampling frequency. For unsupported WAV files, such as 8-bit mono/stereo audio or different sampling frequencies of the two audio samples, the program should display a message saying that the WAV format is not supported and display the following information on both *`f1.wav`* and *`f2.wav`* and then exist.
   a. Number of channels
   b. Number of bits per sample
   c. Sampling frequency
   d. Duration in seconds
4. Users should be able to switch the audio playing through the following keys:
   a. Key 'r' or R': rewind to the beginning of the audio and play
   b. Key 't': tog the looping and non-looping playing mode (default looping).

   c. Key ‘q’ or ‘Q’: exit the program
5. When key ‘t’ is pressed by the user at any time, the program should CONTINUE playing the audio from the current position. In other words, the program MUST NOT rewind to the beginning of the audio when the key is pressed.
6. You MUST supply a *Makefile* together with your source code to compile and link your program.
7. Except the main SDL, no third-party libraries and additional SDL packages such as `SDL_mixer` should be used in the program. The code has to be in C/C++.

Marking Scheme

1. Zero marks may be graded if your code cannot be compiled using the supplied *Makefile*.
2. Program structure, comments and usability (2%)
3. Rewinding, looping and non-looping playing (2%)
4. Generation of the mixing audio signal (6%)
5. Display of information on unsupported audio files (2%)
6. Support of 16 bit mono/stereo WAV files (3%)

Useful References

- SDL Reference manuals on subject Moodle or https://www.libsdl.org/
- Example code (`sample4a2.c`) that raises or lowers the volume respectively by factor of 2 when the key ‘=’ or ‘-` is pressed.

Submission

Zip all source files and the comparison report to ***your_login_name.*zip** and submit the zip file.

IMPORTANT: ***DO NOT*** *include and submit any object files and images in the zip file. Your submission may not be accepted if you do so.*