CMPT 365 Multimedia Systems

Project 1

Deadline 8:00pm, October 6 (Fri), 2023

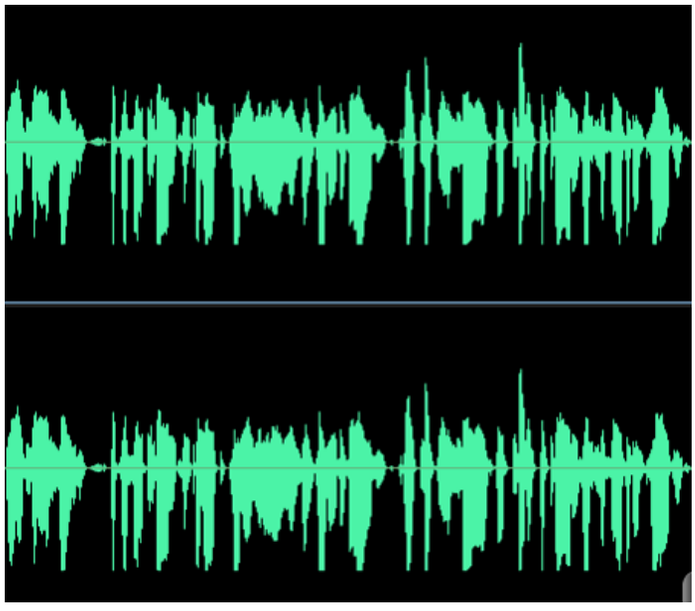
Note: This is NOT a group-based project. Each student should submit a project report and the codes. We will use code scan tools to ensure that the programs are written by yourself. Please don't copy others or let others copy yours --- both will be considered plagiarism.

Later projects will be based on Project 1. So try your best to make the codes work. The programming language allowed will be C/C++/Java only. No Python or others.

Submission guideline: TA will announce detailed requirement soon.

Q. 1. Create a program that reads a .wav files, and then plot the waveform of each channel on the screen.

Your program should first show an "open file" dialog box for loading the wave file. Your program should then display the waveforms on the screen (see a sample below for stereo sound). Also show the total number of the samples and sampling frequency on the screen.



You can Google the .wav format. We assume that the audio file is PCM coded and stereo only (i.e., no mono or multi-channel beyond 2), and the total number of samples will be an even number less than 65536. Some .wav file examples will be supplied later, and the files for our test will be similar to them.

You can call any existing library function that automatically interpret the .wav file or write the function by yourself. For displaying waveforms, you have to draw them using basic pixel/line plotting operations in C/C++/Java), i.e., you can't call an existing library function that directly outputs waveforms.

You should submit your source code and executable code, together with a report that includes screen shots of open file dialogs and displayed waves of the sample inputs, as well as the key GUI functions you have used.

Q. 2. Create a program that reads an uncompressed .tif image file and displays it on the screen. Only the 24-bit RGB full color uncompressed mode in TIFF will be considered. You can assume that the image is no bigger than the 4*CIF size (i.e., 704*576).

Your program should have a menu item “Open File”, which, once clicked, will show an open file dialog box for the user to select a TIFF file. Once selected, the program will then display the TIFF file. The user may then use “Open File” to select other TIFF files to be displayed. Each old image should be wiped off when displaying a new image. Note that the images may have different sizes. There should also be a menu item “Exit” for terminating the program.

You should submit your source code and executable code, together with a report that includes screen shots of open file dialogs and displayed sample images, as well as the key GUI functions you have used.

Rationale of doing this project:

1. Practice GUI, which is necessary for multimedia applications (and indeed, most modern computer applications)

2. Learn a media file format and how is it specified. You will find that there are so many fine details. There are so many different formats in industry (see for example *Murray, James D.; vanRyper, William (April 1996). "Encyclopedia of Graphics File Formats" (Second ed.). O'Reilly. ISBN 1-56592-161-5.*) Yet as long as you have experience with one format, you’ll find that you can easily hack most others. TIFF is a commonly used (and powerful) image format.

   If you want to have hands-on experience, we suggest that you read the TIFF format description and write a bit-level parser to decode the input file. This can be challenging but will be very rewarding. Anyhow, if you don’t want to take this challenge, you can Google existing libraries. If you write the parser by yourself, please do highlight it in your report. If not, list the libs you used to read and parse the TIFF file.