ISTE Audio Processing SMP – Assignment 1

**Q1**. Given the sample rate, bit depth and duration of a song, how would you calculate (approximately) the size of the song? (in megabytes)

The sampling rate sr is the number of samples takes per second.

Each sample takes up memory in bits equal to the bit depth bd.

Thus, the memory taken up by an audio file per second in bits is the product of the sampling rate and bit depth i.e. sr\*bd.

If the file is *t* seconds long, then the memory taken up by it in bits is sr\*bd\*t.

In megabytes, the memory is given by **sr\*bd\*t/8388608.**

**Q2**. Sometimes when you are learning to play a song on an instrument, it can be useful to slow the song down in order to hear some parts more clearly. How can you theoretically change the playback speed of an audio file?

To change the playback speed of an audio file, a higher or lower rate can be specified in the Audio attribute of the IPython.display module.

import librosa

import IPython.display

audio\_file = 'Original.wav’

x, sr = librosa.load(audio\_file)

IPython.display.Audio(x, rate = sr/2) # Lower rate sr/2 slows down the song. It becomes twice as long.

**Q3**. Write and execute a code to change the playback speed of the audio file provided. Increase it to twice the original speed and also decrease it to half. Save the audio as a new file in each case. (Do not use any new functions for this)

import librosa

audio\_file = 'Original.wav'

x, sr = librosa.load(audio\_file)

print(x.shape, sr)

librosa.output.write\_wav('Slower.wav', x, int(sr/2))

librosa.output.write\_wav('Faster.wav', x, sr\*2)