

Music 171: Computer Music I
Assignment #3,
Due: Friday October 25, 2019

In this assignment, you will make a Pd patch that can play a 12 note chromatic scale (above and below the original pitch or your sample) using a saxophone wavetable loaded from the wavefile `SopSax.NoVib.ff.C4`.

Play a single note from your scale by selecting a number, between -12 and 12, that specifies the number of semitones above/below the original pitch. Since the phasor repeats, make sure to turn on/off (by scaling amplitude) using your calculated sample duration. Transposing the wavetable will change its duration (e.g. 12 semitones above the original pitch is a transposition factor of 2 and will half the original duration). Make sure to turn the note on/off with this new duration.

BONUS: use the counter (from previous assignment) to play the scale automatically.

1. Read the wavefile to an array (remember the resize flag) using the `soundfiler` object. The output of `soundfiler` will give you the length of the wavefile in samples N and the sampling rate f_s which you can use to calculate the original sample duration.
2. Take the inverse of the duration to calculate the original (untransposed) phasor frequency f .
3. For an initial test, listen to sound at original pitch/duration:
 - (a) put f into the left inlet of `phasor~`;
 - (b) multiply output of `phasor~` by N ;
 - (c) use output of multiply as (left) input to `tabread4~` (with array name as argument);
 - (d) multiply output of `tabread4~` by on/off values (initially set to on if you want to hear your sound) and input to `dac~` object.
4. Change pitch:
 - (a) use a number box h to specify semitones above/below original pitch;
 - (b) divide h by 12 and put result into right inlet of `pow` object;
 - (c) the left inlet of `pow` should have a message box with value 2 that is “banged” every time a new value of h is produced, outputting a transposition factor t ;
 - (d) multiply t with original phasor frequency f and make this the new (left) input to `phasor~`.
5. Update new duration, and turn on/off automatically (so it doesn't repeat/loop)
 - (a) invert transposition factor t (using `pow -1`) to produce duration factor d ;
 - (b) multiply d with original duration and 1000 (to produce milliseconds) and input (left) into `delay` object; use output of delay to turn sound off;
 - (c) connect h to message that turns sound on, as well as to a 0-message that reset's the phasor's phase.
6. Try BONUS using counter from last assignment (in addition to, and not in lieu of, the previous steps) to automatically play a 2-octave scale.

Submit on TritonEd.