MUSI 2526: Intro to Audio Tech II
Spring 2024

Midterm Project

You and your partner (teams of ~3) will explore the topics of signal processing, synthesis, Python programming, and the various topics covered in this class so far. In this project, you will create a simple electronic instrument/synthesizer. You will give a demo/performance through this system. The system will be implemented in Python, and you will have a Jupyter notebook with all the code to run your instrument. (Code will go through plagiarism review. Be sure to write your own code).

The focus of this project is the implementation of the project—not the performance per se. The performance/demo will be a proof of concept, though points are awarded for conceptual creativity. (However, 10% of your grade will come from you having a working demo for class as well as the ability to explain how your instrument works to the class.) You are **required** to program/implement the following:

- Basic underlying sounds generated by:
    - Simple oscillator function(s) (e.g., sinusoids) and/or your own predefined oscillator functions with customized harmonic spectra. These should output relatively complex and harmonic-rich waveforms (i.e., they should have relatively clear pitch and interesting timbres). **OR, optionally:**
    - the modification of sampled sounds (monophonic, normalized, and set to int16) as unit generators with pitch shifting. Sampled sounds should also be relatively complex and harmonic. You may record the samples yourself or use preexisting musical samples.
- Should be able to pass to your digital instrument a series of symbolic pitch values (e.g., scale degrees, note names) *not* frequency in Hz. (MIDI values are discouraged but accepted.)
- Use of 3+ of the following *in combination*:
    - AM synthesis
    - FM synthesis
    - Pulse Width Modulation (PWM)
    - ADSR envelope
    - Low-pass, high-pass, or bandpass filter
    - “reverse tape” technique
    - Delay

You are encouraged to “compartmentalize” your functions and chain functions together. You are also encouraged (read: challenged) to come up with a unique musical performance (i.e., You should not ‘play a note or two’ with your synthesizer but have—at a minimum—a melody, chord progression, or a combination of both. For instance, you may attempt to emulate a particular instrumental sound, or you may wish to try to replicate a portion of an existing piece of music (electronic or popular song, film music, etc.) You are encouraged to experiment with as many ‘patches’ as you like. Your in-class presentation will be limited to three (max) with a minimum of two.

Deliverables:

- All code used (Jupyter notebook) as well as uploads of any sampled sounds you generated yourself so that instructors can accurately run your notebook.
- Code should be commented in markdown or with comments to explain logic and functionality clearly.
- Schematic block diagram of your system (any image format, e.g., png or pdf). May be hand-drawn *so long as it is easily legible*.
- A 10 minute presentation and performance/demo (Thurs, March. 7).

Grading:

- Requirements met (completeness/functioning code) & deliverables /20
- Effort /20
- Creativity and Musicality /20
- Demonstrates understanding /10
- Presentation/Demo (Explanation of creative process, choices *and* functionality) /10
- Efficiency, documentation (and legibility) & functional organization /10
- Peer-to-peer learning (group/self assessment) /10