**AT 101: Creative Codes**

**Fall 2024**

**Ammerman Center for Arts and Technology Connecticut College**

Daniel Fishkin / dfishkin@concoll.edu / dfiction.com

SYLLABUS IS SUBJECT TO REVISION DURING THE SEMESTER

*Course Description:* This introductory course presents a wide range of interdisciplinary art practices - music, visual art, and poetry - and contextualizes them with (and against) paradigms in computer programming. By focusing on thematic resonances, such as recursive structure, physical computing, and object-oriented programming, the intent of this course is for students to create meaning in working with code, rather than approaching computers as a default solution to every problem.

This course provides a foundational introduction to music programming languages. Max MSP, SuperCollider, Python, and Processing and Open will be explored through hands-on and creative applications. Students will gain experience with interactive visual art, audio/music signal processing, and machine learning.

This course does not provide a thorough introduction to any single computing language. Rather, it expects an encourages a humanistic approach translating higher level concepts of art to lower and very low levels of applications specific to your intent.

*Lecture Time:* Tuesdays, 7:00 pm — 9:40 pm

*Lecture Location:* Cummings 227 (Manwaring Computer Lab) / Online as needed

*Instructor:* Daniel Fishkin

*Office Hours:* **by appointment (IRL or Remote)**

*E-mail:* [daniel.fiction@gmail.com](mailto:daniel.fiction@gmail.com) / dfishkin@conncoll.edu

*Materials Fee:* $50

*Textbooks:* Valle, Andrea. *Introduction to Supercollider. Logos Verlag Berlin. 2016. 3832540172*

Manaris, Bill. *Making Music with Computers: Creative Coding in Python.* Routledge. 2014. 1439867917

Shiffman, Daniel. *The Nature of Code. Creative Commons. 2012. 0985930802*

Software **Max**

<https://cycling74.com/>

Max is an object-oriented programming language that makes easy prototyping of sound environments/instruments accessible without needing to learn to code on levels like JS or C. It is a type of coding language in its own way, but the coding is done primarily with virtual representations of wires. It’s very fun and many of my class demos will get you excited about what’s possible—download it and see what you can get working off the bat!

*Note: you can try Max for free for 30 days, and thereafter you’ll need to purchase it, or sign up for a monthly or yearly student subscription. It is affordable. Conn College has Max on every machine in the computer lab!*

**Max comes with an extensive body of educational materials, tutorials, help files, and project examples that can easily be copy/pasted into your own sketches.** Students will be expected to read and study Max / MSP tutorials on their own and learn the syntax of the software throughout the course of the semester!

**Enter the MAX file browser and search:**

**collection:"Tutorials/Max Tutorials@cycling74"**

**collection:"Tutorials/MSP Tutorials@cycling74"**

**Supercollider**

<https://supercollider.github.io/downloads.html>

Supercollider is a platform for audio synthesis and algorithmic composition, used by musicians, artists and researchers working with sound. It is code-based, completely free, light in size, and a little hard to work with.

**Spear**

<https://www.klingbeil.com/spear/>

Spear performs Fourier Analysis in sound files. This is free software that allows you to turn recordings into individual sine waves and edit/select them.

**Audacity**

<https://www.audacityteam.org/>

Audacity is a basic editing program that is excellent for transforming individual sounds.

**Python**

Extremely powerful code based platform

<https://www.python.org/downloads/>

Some possible code editors you could use:

**Visual Studio Code**

<https://code.visualstudio.com/>

Pretty! Organized.

**Sublime Text Editor**

<https://www.sublimetext.com/>

Simple. Easy.

**Csound**

[*https://github.com/dfict/clawyer*](https://github.com/dfict/clawyer)

A rather old school way of making sound with computers.

Hardware: <https://www.frommdesign.com/product/ad-1-dsp-dev-board>

AD-1 is a DSP development board based around an ESP32-s3 microcontroller and an es8388 audio codec. Designed for experimenting with digital audio, the AD-1 can generate sound on its own or manipulate incoming stereo audio. This board costs around $50 and will be the basis for many hardware projects.

**Grading Rubric**

Class Participation / Attendance 20%

Weekly Corpus 20%

Etudes / Projects/ 20%

Presentations 20%

Final Project 20%

**Grading Descriptions**

**Class Participation / Attendance 20%**

Class attendance is mandatory. We work and experiment in class. Discussions, critiques lectures and demonstrations provide the basis for the successful completion of projects, and they are difficult to re-create outside of class. In order to participate, you must be in attendance. You have one unexcused absence permitted for the semester. Your final grade will drop by 4 points for each further unexcused absence. More than four absences will result in a failing grade for the course. You are expected to participate actively in class by asking questions, bringing energy to discussions, and arriving with prepared homework/projects. Independent motivation is expected.

**Weekly Corpus 20%**

Students are expected to keep a weekly audio “journal”. Every week, students will collect/record/curate sound file collections. The collections each week will total up to 5 minutes of sound (or more), featuring at least 5 different recordings. (**ie**, Week 1, you record five 1 minute recordings. Week 2, twelve 10 second recordings, and one 3 minute recording). It is expected that you carefully label these files and keep good file management, in additional to normalizing audio and trimming the beginning and endings of each file. You can record with a high quality field recorder you borrow from the university, you can record with your phone. You can also “sample” recordings found in the wild (but are discouraged from sampling pop music.)

Some prompts will be provided to help simplify this task. But ultimately you are encouraged to: **make it your own** and may disregard time and file requirements as the semester goes on and you figure out what your’e doing.

**Etudes / Homework 20%**

Short reading assignments or creative prompts will be assigned each week. Students will be expected to complete them and come to class ready to discuss their progress. You cannot “get ahead” of these assignments by doing them all ahead of time—they are meant to be part of a weekly practice that is customized to the flow of the class.

**Artist Presentation 20%**

Do a short research presentation on an artist or piece of art that inspires you. 10 min. The purpose of this assignment is to aid the development of your final project by creating an opportunity to research something that interests you for your own creative purposes and research agenda.

**Final Project 20%**

Make a creative thesis and execute it, utilizing your audio corpus. The project may include video, audio, drawing, and must somehow utilize a computational process to illuminate your corpus.

**AI Policy**

The use of Machine Learning tools such as ChatGPT and Claude are permitted, generally. In some cases we will explore them deliberately. In some instances you will be discouraged from using these tools. AI represents a sea change for humanity. It also represents a paradigm shift for pedagogy in digital literacy.

I expect you will want to use AI (ChatGPT and image generation tools, at a minimum), in this class. In fact, some assignments will require it. Learning to use AI is an emerging skill—be aware of its limits. If you provide minimum effort prompts, you will get low quality results. You will need to refine your prompts in order to get good outcomes. This will take work.

Don’t trust any code that your AI provides you. You will be responsible for any errors or omissions provided by the tool. AI is a tool, but one that you need to acknowledge using. Please include a paragraph or citation on any assignment that uses AI explaining what you used the AI for and what prompts you used to get the results. Failure to do so is in violation of the academic honesty policies.

*SCHEDULE*

*Class 1 Tuesday 8/27*

**Introductions**

Syllabus Review

*Student Introductions*

Art from Pre Computer Algorithms

Sol Lewitt — pencil instructions for drafters

Tristan Perich (drawing machines)

Basic Concepts in Code: conditional logic, boolean statements

Introduction to Acoustic & Electronic Sound

Introduction to MAX MSP <https://cycling74.com/>

**Begin playing with Max!**

**Homework:**

Read: What is Code? By Paul Ford

<https://www.bloomberg.com/graphics/2015-paul-ford-what-is-code/>

We will discuss next week.

Write your own code poem or drawing:

PICK ONE:

1) by modifying one of the Montfort examples.

2) by writing instructions ala Sol Lewitt and executing it

3) by writing a prompt for YOUR OWN CODE POETRY IN PSUEDO-CODE…and to execute it.

**First Corpus Prompt:**

Record sounds from your life. At least one sound should be a drone. They should not be boring sounds. Don’t record something like typing on your computer. Don’t waste tape!

Class 2 Tuesday 9/3

**Making Sound**

Basic Concepts in Code: conditional logic, boolean statements

Introduction to Acoustic & Electronic Sound

Introduction to MAX MSP <https://cycling74.com/>

**Fundamentals of Synthesis / Tuning**

**Introduction to Control Voltage —automation (modulating pitch, volume and other things)**

Artist Profile: Wendy Carlos / Morton Subotnick

Moog / Bucha / Serge

Reviewing the sounds from your first corpus.

Audacity Tutorial—how to mix, normalize, trim and edit waveforms.

We edited and processed and normalized 5 sample sounds, labeled—they’re in this folder: try to keep track of your files with nice, pretty, organized labeling schemes!

<https://github.com/dfict/CreativeCodes/tree/main/normal_files>

Class 3 Tuesday 9/10

More max (sampling)

Introduction to Acoustic & Electronic Sound

Introduction to MAX MSP <https://cycling74.com/>

**Fundamentals of Synthesis / Tuning**

In class exercise: manipulating max patches and coding from scratch.

**Second Etude**

make a simple synthesizer in max.

it can do anything you want it to do, as long as you’re able to control it. Make sure it is controlled by the buttons ASDFG. Consider: what tuning system should it adhere it? How many voices should it have? should have control of volume AND pitch.

**Corpus 3**

Prepare 5 or more recordings. One recording should contain a voice. The second recording should contain a pitched instrument sound. The third recording should contain a percussion sound. All files should not be too short (less than a second) or too long (more than a minute).

Class 4 Tuesday 9/17

Lecture:

History of sampling

Max Topics:

Sampling, working with soundfile input / microphone input.

In class exercise:

Joo Won Park’s 4 hit combo

**More Code Poetry**

“Psuedocode” approaches: Sheila Heti’s Alphabet Diaries

Real code approaches: Lillian-Yvonne Bertram’s Travesty Generator

Homework:

Class 5 Tuesday 9/24

Max Topics:

**Sol Lewitt revisited**

Amy Goodchild’s wall drawings

<https://www.amygoodchild.com/blog/ai-generations-chatgpt-4-sol-lewitt-wall-drawings>

<https://www.amygoodchild.com/blog/ai-generations-chatgpt-4-sol-lewitt-wall-drawings>

<https://editor.p5js.org/amygoodchild/collections/4lhhLgjzw>

sequencing, fluid corpus manipulation

Class 6 Tuesday 10/1

**No Class Tuesday 10/8—Fall break!**

Class 7 Tuesday 10/15

**Introduction to Supercollider**

Class 8 Tuesday 10/22

Hardware day

Esp32

Class 9 Tuesday 10/29

Class 10 Tuesday 11/5 Election Day!

Class 11 Tuesday 11/12

Class 12 Tuesday 11/19

No class Tuesday 11/26—happy thanksgiving break!

Class 13 Tuesday 12/3

Class 14 Tuesday 12/10

*12/12—12/16 Final Exam TBD*