

Algorithmic Art Syllabus

What is an algorithm? What is it about them that lets us navigate large maps or search the web? Can we use techniques in electrical engineering, computer science, and design to create artistic effects through algorithms? In this course, we'll discover the basics of programming and algorithmic thought by learning the fundamentals of algorithms. This includes important processes like path finding, searching, and packing algorithms. We will tie these back to visualizing these algorithms and their processes with open source software in artistic and novel ways. Using what they've learned, students can make a variety of beautiful visualizations and their own original art pieces through code, electronics, and analog effects.

No programming or artistic experience required, although some is recommended.

Objectives

This course will be taught studio style with a significant portion of time dedicated to student work. This is so students can explore topics that are of particular interest to them and walk away with final projects they feel good about.

Throughout the course the students will gain a knowledge of:

- Algorithmic and rule based thought
- Overview of algorithms that are present in computer science
- A deeper understanding of key algorithms
- Rule based drawings from algorithms or other processes
- Basics of visual programming
- Electronic fundamentals
- Real life applications of algorithms
- Scoping out their own individual projects

Schedule — July 15-20, 9am to 4pm

Day 1

- Algorithmic thought and rule based processes
- Rule based drawing and exercise
- Survey of algorithms in EECS
- Electronics overview and exercise — add an LED to your drawing

Day 2

- Visual Code 101 with a graph
- Sorting Algorithms
- Coding in color: More electronics — adding color to your drawing
- Searching and Pathfinding Algorithms
- Coding a visual path

Day 3

- Packing Algorithms and visualizing packing
- Visualizing line of sight
- Visualizing a moving flock
- Project planning

Day 4

- Electronics and coding hands-on
- Work time
- Peer brainstorm and progress meeting

Day 5

- Work time
- Peer presentations and discussion
- Where to go in the future of algorithms and algorithmic art

Instructor Info

Nina Lutz
Graduate Student, MIT Media Lab
nlutz@mit.edu

Additional Resources

<https://processing.org>

<https://www.codecademy.com>

<https://stackoverflow.com>

<https://www.geeksforgeeks.org>

<https://drawingwithcode.wordpress.com>

<https://sfpc.io>

“Generative Art: A Practical Guide Using Processing” by Matt Pearson

<https://www.openprocessing.org>

<https://creative-coding.decontextualize.com>

<https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-006-introduction-to-algorithms-fall-2011/>