DESMA 1: Social Software Studio — Art & Code

Broad Art Center, Remote
Monday–Friday, 10am–1pm and 2–5pm

This course explores what code can do within the visual arts and what we can do with it. It focuses on exercises with templates that can be modified and adjusted for new learners and that can be extended by people with prior knowledge in this area. A series of short exercises cover a wide range of possibilities. Exercises will be light and short, followed by a final project. We will explore a range of experiences including drawing, coding for the web, interactive performance, and generative art.

We will create software within the realm of the visual arts. We will build a foundation that serves as a platform for future learning. The class is taught as a series of workshops and discussions with demonstrations and time to work.

This course asks a few questions:

- How has software affected the visual arts?
- What is the potential of software within the visual arts?
- As an artist or designer, why would I want (or need) to write software?

Assignments

This class is built around three exercises and a final project. Each exercise has an in-class workshop and a discussion about your finished exercise on the day that it's due.

Evaluation

Grading is based on the Exercises and active engagement. Focus, articulation of ideas, keeping up with the work and assignments, and contribution to class discussions are all part of class engagement. All work will be evaluated based on (1) the fundamental idea you develop and (2) the craft, meaning the details of the images, motion, and interaction.

The numeric breakdown for all assignments follow:

15% — Exercise 1
15% — Exercise 2
15% — Exercise 3
35% — Final Project
20% — Engagement

I have the strong expectation that you will join for each class and workshop session. If something happens and you can't join the class, please email me. If you feel frustrated or you come across other problems, please communicate with me directly and quickly.

Commitment To Equity And Diversity

We understand the classroom as a space for practicing freedom; where one may challenge psychic, social, and cultural borders and create meaningful artistic expressions. To do so we must acknowledge and embrace the different identities and backgrounds we inhabit. A collaborative effort between the students and the teacher is needed for creating a supportive learning environment. While everyone should feel free to experiment creatively and conceptually, if a class member points out that something you have said or shared with the group is offensive, avoid being defensive; instead approach the discussion as an opportunity for everyone to grow and learn from one another. All class members are encouraged to discuss such instances with me so they can be addressed with greater care in the future.

Media

We'll cover everything you need to know in class, but additional media can be useful for further exploration and reinforcement.

- Intro to Programming in p5.js by Xin Xin
- Getting Started with p5.js. Lauren McCarthy, Casey Reas, Ben Fry
- Generative Design: Visualize, Program, and Create with JavaScript in p5.js. Benedikt Gross, Hartmut Bohnacker, Julia Laub, Claudius Lazzeroni

Exercise 1 — Conditional Drawing

Use the drawing exercises from Workshop 1 and your own knowledge of art as a reference point to create your own instructions for a unique drawing system. In total, someone should be able to make a drawing from your instructions within fifteen minutes. This is a solo project, each person in the class will create their own instructions.

There are two parts to this: 1) invent a drawing system 2) encode the system as text. To work through both parts, I recommend starting by sketching. Try many things out quickly and fluidly. Once you have some sketches that feel promising, start to write the instructions. Ask someone to create a drawing from your instructions and evaluate the result. Did the drawing turn out as you expected or not? If not, take what you learned to modify the instructions and try again. This should be an iterative process. Push things and don't be satisfied with the first results.

Notes:

- Think about your drawing materials, marking-making tools and drawing surfaces

- Think about different types of marks that are possible with analog materials that might not be possible with digital materials
- Do you want the drawing to incorporate tools like a compass or ruler? Or anything else?

Exercise 2 — Digital Painting

Today we experienced Kid Pix as one example of thinking about software drawing tools as unique ways of making marks without the constraints of physical tools such as pencils and brushes. Think about your experience and create a weird software drawing tool in p5.js to draw a specific kind of thing. For example, create a drawing tool for hair or to render mutated plants. Pick something that's unique to you and you will enjoy creating.

This is a group project, you may work with one other person.

You're not making a full drawing system; focus on something narrow and possible to do for next Tuesday. Focus on one (or a few) custom brushes. Start by sketching on paper or digital drawing software, then move into writing code.

Create five original drawings with your software and save them as PNG images with save() inside of the keyIsPressed() event function. Add the drawings to your code folder when you upload it to the server. We'll look at the drawings in our discussion on Monday.

Here's a template to get started with:

[Code sample to come…]

At minimum, use the functions we learned including setup(), draw(), mouseX, mouseY, mousePressed, keyIsPressed(), and if within your code. Expand as you wish.

Notes:

- You are making a drawing tool to draw something specific, not a general-purpose drawing tool that can draw anything in a generic way. Make it custom for your unique idea.
- I recommend using the keyboard to control the options for your drawing program rather than creating a graphical toolbar. Making a graphical user interface (GUI) is difficult and we'll work our way there over the weeks. Write a substantial comment at the top of your code and remember to edit and spell check.

Tips:

- Take your time and be careful
- Format your code with spaces or tabs so you can see the structure
- Take a break when coding gets frustrating
- Start the project early, well before the deadline

- Ask questions, please!

Exercise 3 — Generative Drawing

Create a generative drawing p5.js sketch in response to what we looked at today. Do your own thing, going with or against what you see happening within those references.

Artists have been working with code since the 1960s. The last two years have seen the largest surge of interest within this kind of work within its history. We've also seen a surge in artists working with code to create generative representational landscapes and illustrations. What is your take on this? How would you like to respond?

Here's a template to get started:

[Code sample to come…]

Notes:

- This is a short experiment. Be clear about what you want to achieve and work toward that step by step. You probably won't have enough time to do everything you want.

Tips:

- Take your time and be careful
- Take a break when coding gets frustrating
- Work together and help each other
- Ask questions, please!

Final Project — Mask

Use the examples we covered in class as the technical foundation for creating a digital "mask" or "costume." When one or more people are within a video image, replace their faces and/or clothing with digital images. You can approach this as a performance system, a critique of vision systems, or a visual aesthetic experience, among many other options. The aim is to think more about machine learning and computer vision systems through creating your own software that utilizes the technology.

This is a group project, you may work with one other person!

Code examples

- Facemesh (Reference, Mesh Map)
- PoseNet (Reference)
- FaceApi (Reference)