a-generative-web

instructor: john provencher
provj824@newschool.edu
(1/25 — 5/10) 2023
wednesday: 12:10pm - 2:50pm
location: 66 5th ave room: 824

description

in this course, students will use the generative nature of the web as a medium to shape work(s). students will learn generative approaches to make visual and conceptual systems by programming in javascript … students will adopt a generative mindset: to create scripts that interact with varied conditions to produce an array of outcomes rather than fixed ones. students will study generative works by evaluating the past and present of graphic design, art, writing and new media.

structure

the class will meet once a week for 15 weeks over the course of the semester. there will be 1 assignment and 2 projects. class time will consist of lectures, workshops / demos, general discussion and reviews. there will be a lecture that gives context to each project. the workshops / demos will be a time for students to start learning the basics of generative javascript programming. reviews will consist of 1-1s, small groups and class review. each project will have a final review. media/readings will be assigned periodically and students are encouraged to bring thoughts, questions and critiques to discuss.

notes on programming proficiency

a basic understanding of html/css/js will be recommended for the class, as well as an ability to deploy a basic website. 2 projects will be programmed in javascript and, although this course presents many technical challenges, students should expect to do their own research outside of class to fill in the gaps of programming knowledge for their own project.

a1
letter [8.5x11]
a written prompt for a drawing

time: 1 week
input: a listed prompt written to create a drawing.
output: a drawing from each student in class.

- how is/was generative art made without code? how do we create unbiased variation from the human hand?

- in this project, students will create an instruction for a drawing on letter sized paper.

- the drawing can be anything you'd like but it should not take more than 5 minutes to make.

- each instruction will be drawn by other students, so students have to design a simple set of instructions that can be clearly followed.

- the language of the prompt should embrace randomness over replica, enabling the participant to come up with intuitive decisions without confusion.

p1
canvas [*100]
a generative canvas

time: 8 weeks
input: a script written in javascript that uses rng to create a variety of outputs.
output: a poster, painting or image that has at least 100 variations. students will create a html website with the working display of their script and at least 100 archived images.

- treat the web as a <canvas> ... is it for a painting? for a poster? for a world?

- in this project, students will propose an image-making script that generates at least 100 unique outcomes on a <canvas>.

- the proposal should include research, concepts and sketches that help narrow in on an idea for the script.

- the project can take on a broad array of concepts so students will have to explore and research as they develop their script.

- students are encouraged to think of the context that their work lives in, on and off screen.

- the script itself should have at least 100 unique outcomes with metadata.

p2
interface [∞]
create a 'character'

time: 6 weeks
input: a program written in html/css/js that is controlled by using a gui interface
output: a web tool that creates an almost infinite range of 'characters'.

- the create a character page is my favorite part of a videogame. each slider and input gives an infinite amount of possible outcomes.

- for this project, students will propose a 'create a character' tool that outputs varied outcomes of an interpretation of 'character'.

- though the prompt heavily suggests a literal interpretation of 'character', students are encouraged to think outside of the literal definition by defining what character means to them and how to approach it (typographic, abstract, animation, 2d, 3d, pfp, identity, etc).

- similar to the past project, the proposal should include research, concepts and sketches that help narrow in on an idea for the script.

- this project will rely heavily on the web interface that controls the variables of the ‘character’. students will be designing a fully functional interface as well as the subject character.

required tools

- a browser
  (chrome, firefox, safari, brave)
- a text editor
  (sublime text, vs code etc.)
- figma & or adobe suite
- github desktop
- node.js
  - (https://nodejs.org/en/)

resources

libraries

- http://paperjs.org/
- https://p5js.org/
- https://threejs.org/

programming assistance

- https://www.freecodecamp.org/
- https://www.codecademy.com/
- https://www.learn-js.org/
- https://thecodingtrain.com/

grading assessment criteria

attendance, grading and work submission standards, program policies, making resources, and university policies. all cd classes adhere to the same program and university policies: (link)

10%	participation
10%	a1
40%	p1
40%	p2

learning outcomes

- practice generative and computational art / design
- practice prototyping generative systems and interfaces
- practice planning, designing and publishing websites
- research history and context for the www and how artists/designers use it as a medium.
- research material to help with self-learning programming.
- research current design, art, and tech precedents.
- chances to give and receive feedback with an open, generous and constructive mindset.
- chance to combine personal artistic creativity with class assignments

log

the schedule of projects, assigned media, and async work can be seen below. to see a more complete overview each day, refer to a-generative-web.com which will be updated each week.

01.25	
class	● introductions ● a1 lecture ● demo: paper.js, http-server
media	● grapefruit-ono.pdf ● conditional-design-moniker.pdf ● conversation-with-john-cage.pdf
async	assignment 1: ● due: 16 printed instruction sheets with your instructions

02.01	
class	● assignment 1 final review / exercise ● p1 assigned, ● demo: random()
media	● designing-programmes-gerstner.pdf ● game-design-habraken.pdf
async	project 1: ● due: presentation in .pdf / .key format

02.08	
class	● p1 proposal reviews, lecture ● demo: functions, if / else
media	● jodis-infrastructure-galloway.pdf ● franke-computerart.pdf
async	project 1: ● due: 5 unrelated practice scripts

02.15	
class	● p1 class review ● demo: for loop, grid
media	● weaving-klu¨ck.pdf ● source-arcangel.pdf
async	project 1: ● due: 5 programmed sketches for your project

02.22	
class	● p1 1-1 review ● demo: questions
media	● nfts-readymade-joselit.pdf ● manetas.com/txt/videogamesis.html
async	project 1: ● due: a script that produces 25 unique outputs ● screen-shot 25 outputs

03.01	
class	● p1 1-1 review
media	● form+code-raes.pdf ● history-of-internet-art
async	project 1: ● due: a script that produces 50 unique outputs ● screen-shot 50 outputs

03.08	
class	● p1 class review ● demo : meta-data
media	● automation-understanding-media.pdf
async	project 1: ● due: a script that produces 100 unique outputs. ● 100 outputs with metadata

03.15	
class	● spring break - no class

03.22	
class	● p1 1-1 review
async	project 1: ● finalize project

03.22	
class	● p1 final review ● guest: andreas gysin
async	project 2: ● project 2 assigned

04.05	
class	● p2 lecture ● demo: sliders / inputs
media	● computer agency and behavior
async	project 2: ● due: 3 sketches for outputs ● due: general wireframe for inputs / how they affect outputs

04.12	
class	● workshop: class code skill share ● wireframe overview
async	project 2: ● output refinement

04.19	
class	● p2 class review
async	project 2: ● input refinement

04.26	
class	● p2 small group review
async	project 2: ● prototype a

05.03	
class	● p2 1-1 review
async	project 2: ● prototype b

05.10	
class	● p2 final review

	- guest: tbd
async	class material - send a final email of all class links to: provj824@newschool.edu