

GSND 6460

Generative Game Design

Lecture 0: Welcome

Chris Martens
Wednesday, Jan 8, 2025

Welcome!

- Introductions: me, the TA, you
- Introduction: course topic
- Introduction: coursework; what to expect

About me

- Chris Martens (call me “Chris” or “prof(essor) (Martens)”)
- Pronouns: they/them
- Associate prof, Khoury + CAMD (50/50); at Northeastern since fall 2022
- “Home” = Pittsburgh, PA; Columbus, IN; Austin, TX
- Research areas: Programming Languages; Games
- Disciplinary hats: mathematician, computer scientist, artist, game designer
- Recent joys: bouldering; nonograms; Isles of Sea and Sky



About the TA: Yiming (Jasmine) Sun

Pronouns: she, her

Program/year: Experience Design MFA - 2nd year

1-2 places lived the longest: Shandong, China; San Francisco, CA

Research/course-related interests:

- Game UI design
- XR Design & Education
- Healthcare Design

Something that brought joy: Gym (boxing & climbing), Illustration, Reading, Movies

Office hour: 9 am -11 am Wednesdays, Teams or Zoom

About You

Say hello to a neighbor (or two):

- Name, pronouns (optional)
- Degree program/year
- 1-2 places you've lived the longest?
- What has recently brought you joy?

Warning: you may be quizzed on your neighbor's responses :)

Course syllabus

The “working syllabus” is the website:

<https://www.khoury.northeastern.edu/~cmartens/Courses/ggd-s25/index.html>

(I’m behind on class prep!)

This is **NOT** a “generative AI” class

- Algorithmic Art
- Procedural Content Generation
- Parametric Design
- Variational Programming



- Large language models
- ChatGPT
- Stable diffusion
- etc.



Variational Design

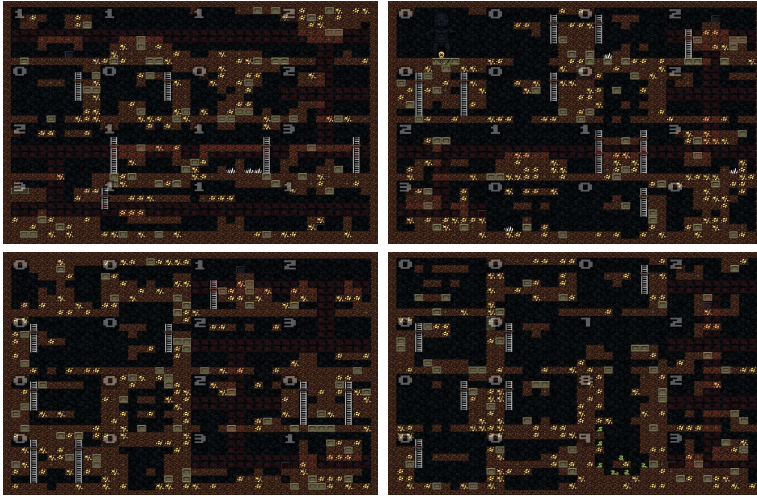
What is it?

- Defining this is a learning objective for the course!
- Broad definition, to be refined:

Variational design is any process of designing many variations on an idea, rather than just a single instantiation.

Examples

1. Procedural generation (PCG): a computer program that generates roguelike levels (below: Spelunky)



2. Procedural art: 100 paintings that all follow the same rule. (Below: Sol Lewitt, Wall drawing #766)

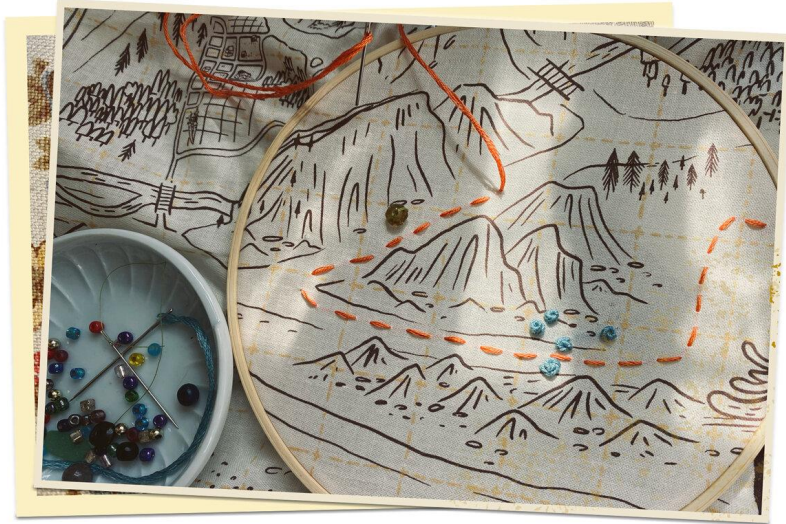


<https://tinysubversions.com/spelunkyGen/>

<https://massmoca.org/sol-lewitt/>

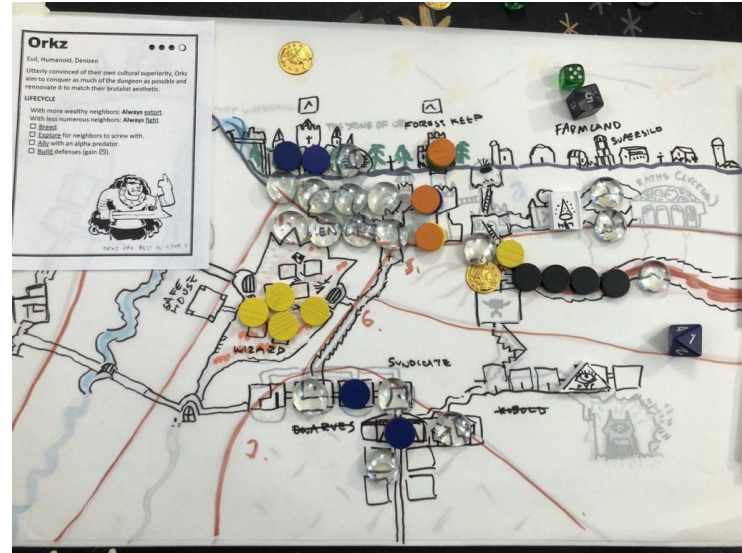
Examples

3. Keepsake games



<https://shingkor.com/installation#/a-mending/>

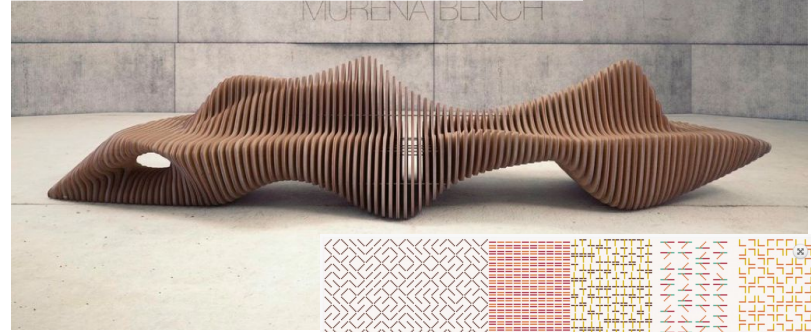
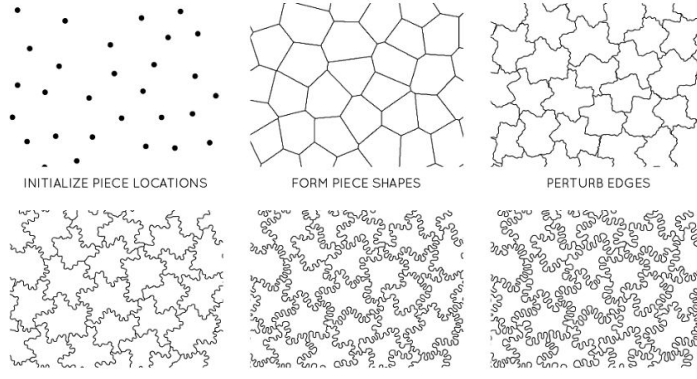
2. Pen and paper/dice table games



<http://www.howtohostadungeon.com/>

Examples

- Pen plotter art
- Parametric design
- Generative design for puzzles, toys, jewelry
- <https://n-e-r-v-o-u-s.com/>
- Generative textile designs
- Generative music / livecoding



WolframTones

Created 2005

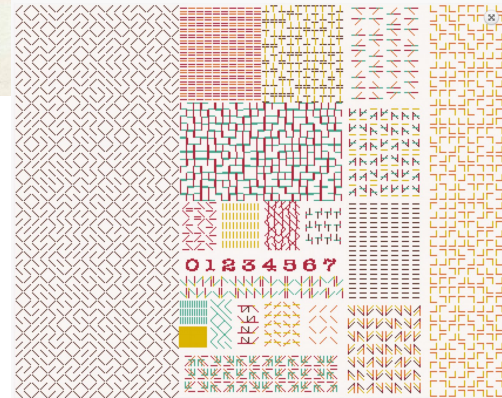
An Experiment in a New Kind of Music — made possible by the Wolfram Language and A New Kind of Science



0:00 / 0:15 Play

Download:

Share:



Variational Design

Values:

- **Artisanship**: programming as craft
- **Intentionality**: making things in service of explicit design goals
- Developing sound **mental models** of code and its behavior
- Computation as an expressive **material** (rather than as a tool)
- Computation as a **medium**: mediating human/human interactions

Two ways to parse “generative game design”

Generative (game design): using generative methods (e.g., PCG) to create artifacts for games (e.g. levels, items, characters, quests)

(Generative game) design: designing games that, when played, generate artifacts

Software/materials/tools for this class

Two main programming languages:

- p5.js (Javascript): <https://p5js.org/>
- Dusa: <https://dusa.rocks/> (experimental!)

Also:

- Paper prototyping materials (dice, cards, pens, paper, etc.)
- Media and tools of your choice! (e.g.: Unity, Haskell, 3D printing, crochet, diorama, ...)

Software/materials/tools for this class

Theme for this year: “where physical meets digital”

Potential collaboration:

- Megan Hoffman, automatic machine knitting:
<https://actlab.sites.northeastern.edu/research/design-tools-for-automatic-machine-knitting/>
- Other possibilities with makerspace resources: board/card games, 3d printing, sewing, laser cutting, papercraft, etc. (let me know what sounds interesting to you!)

Computational topics

- Grammars
- L-systems
- Markov chains
- Rewrite rules
- Logic programming
- Graph search

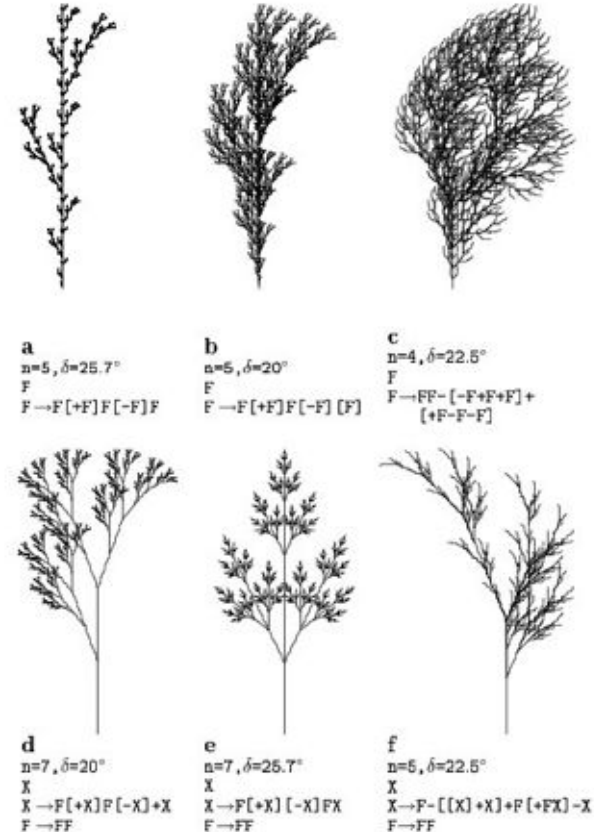


Figure 1.24: Examples of plant-like structures generated by bracketed OL-systems. L-systems (a), (b) and (c) are edge-rewriting, while (d), (e) and (f) are node-rewriting.

Expected student work

input / output / reflection

Expected student work

A balance of **input** and **output**, mediated by **reflection**

Expected student work

A balance of **input** and **output**, mediated by **reflection**

- Possible inputs:
 - Games (play them)
 - Programs (run them)
 - Readings (read them)
 - Your peers' work
- Possible outputs:
 - Sketches (small p5.js programs)
 - Playable prototypes (mostly analog, can be digital)
 - Written reflections

Learning objectives: what will a good grade in this class mean you can do?

Develop small programs ("sketches") that demonstrate generative design techniques.

Explain the concept of a **possibility space** and how a computer program can represent one.

Choose appropriate **data structures and algorithms** for implementing generative methods.

Identify **abstraction boundaries** in generative design problems, choose programmatic representations at the appropriate level, and write programs that **translate between different representations**.

Interpret open-ended creative prompts and deliver **prototype implementations** on a short time frame.

Critique creative work, and have their creative work critiqued, by and for their peers.

Summarize the philosophical approaches of historical and living practitioners of generative design.

Prototype a game that embodies a generative process.

What I want:

- To help you become a better programmer
- To help you become a better designer and “design thinker”
- To find out where your interests meet my expertise, and help you develop them
- To facilitate an environment that is collaborative, not competitive, with your peers, and where you can learn and grow from each other as a cohort
- To learn from *you*

What I want from you

“Care in, care out”

- Show up to class
 - Value face-to-face time for what it's good for
- Do your best to meet deadlines
 - maintain the “rhythm” of the course
 - help my and the TA's workload stay predictable
- Playful curiosity, fun, experimentation without consequences!
- Struggle with ideas – get things wrong, try again, get them right!

What do you want?

Think, pair, share:

- 2 minutes to think/write for yourself
- 5 minutes to share and discuss with your neighbors
- 5 minutes to share out (I'll call on someone at random)

Classroom policies

I have to decide about things like

- What work to assign, on what schedule
- How to give you grades (ugh)
- Whether to try to stop you from “cheating”/using ChatGPT for all your work
- Attendance policies
- Late work policies

We can decide this together (somewhat)

Purpose of today: sort out our shared values/goals

What's next?

- Communication tools: Piazza and Gradescope
- Your first “assignment”: complete an intake form, info about yourself

Next class:

- Intro to p5.js (here are [my notes from last year](#))

Looking ahead:

- I am OOO Jan 20-24 (no class meetings that week)