

CREATIVE CODING

with p5.js

perlin noise by yasai
<https://openprocessing.org/sketch/494102>

Reminder

Week 02: 01_AbgabeInfo.pdf

Website

- Kategorien
 - Shapes / Pattern
 - Image manipulation
 - Etc.
- Vorschaubild pro Kategorie
- Baut eine «coole» Website!

Schlusspräsentation

- 10-30 Minuten
 - Variiert je nach Anzahl Schüler. Mehr Infos später
- Webseite zeigen (inklusive Übersicht)
- Sketches pro Kategorie aufzeigen
 - Interessante Entdeckungen / Algorithmen?
- Individuelle Arbeit aufzeigen
 - Zuerst: Implementation abstrakt (Erklärung ohne Code)
 - Dann: Code durchgehen und erklären
- Wo habt ihr euch Inspiration geholt? Gefällt euch ein Künstler? Motive? Hobby? -> Aufzeigen!

Reminder

Week 02: 01_ AbgabeInfo.pdf

Abgabe

- **VornameName.zip** mit
- Dem Projekt als **Ordner**
- **Anleitung.pdf** wie man das Projekt bei sich laufen lassen kann
 - Link zu [Github](#) / [Gitlab](#) / [Bitbucket](#) / ... (public!)
- **Projektbeschreibung.pdf** vom individuellen Projekt
 - Erläuterung und Erklärung, wie der Code dazu funktioniert
- **Praesentation.pdf**
- **Video.mp4**: Dauer 1min – 5min
 - [FullHD](#), .mp4

Careful

Tight program on the last day:

08:10 - 08:15	Installieren, aufsetzen
08:15 - 08:30	Präsentation 1
08:30 - 08:45	Präsentation 2
08:45 - 09:00	Präsentation 3
09:00 - 09:15	Präsentation 4
09:15 - 09:30	Präsentation 5
09:30 - 09:45	Präsentation 6
09:45 - 10:05	PAUSE
10:05 - 10:20	Präsentation 7
10:20 - 10:35	Präsentation 8
10:35 - 10:50	Präsentation 9
10:50 - 11:05	Präsentation 10
11:05 - 11:20	Präsentation 11
11:20 - 11:40	BUFFER

Hand in **deadline**

The Sunday before the last course day at 23:59!



Deadline ZIP-File

Presentation

Missing the deadline will result in consequences such as additional assignments!

Expected output

- 01:
 - Two starting sketches based on color / random / interactivity / shapes / patterns / strokes
 - At least one own sketch based on what you learned so far
- 02:
 - Your own image filter
 - Sort image pixels by ...
 - Resize image based on interactivity
 - Combine images
 - Image glitch effect
 - Random dithering effect
- 03:
 - Code with the shape looking like a “pipe”
 - Individual agent based on randomness and state changes
 - Implementation of abstract algorithm (Circles)
 - Experiment with collision/shapes and forms

Expected output

- 04:
 - L-System
 - With minimal ruleset for 'F', 'f', '+', '-', '[' and ']'
 - Implement of at least four fractals in your L-System
 - Create/Modify two L-Systems (Colors, StrokeWeight, Interactivity, Randomness)
- 05:
 - Markov chain generator with shakespear.txt as input
 - Wordcloud
- 06:
 - Implementation of conways game of life
 - And another game of life with your own creative twist
- 07:
 - Class challenge

Amount of sketches

~24

Expected sketches

20+

You can leave out up to 4 of the previously yellow-marked sketches.

Reminder

Week 02: 01_Info.pdf

See next slides!

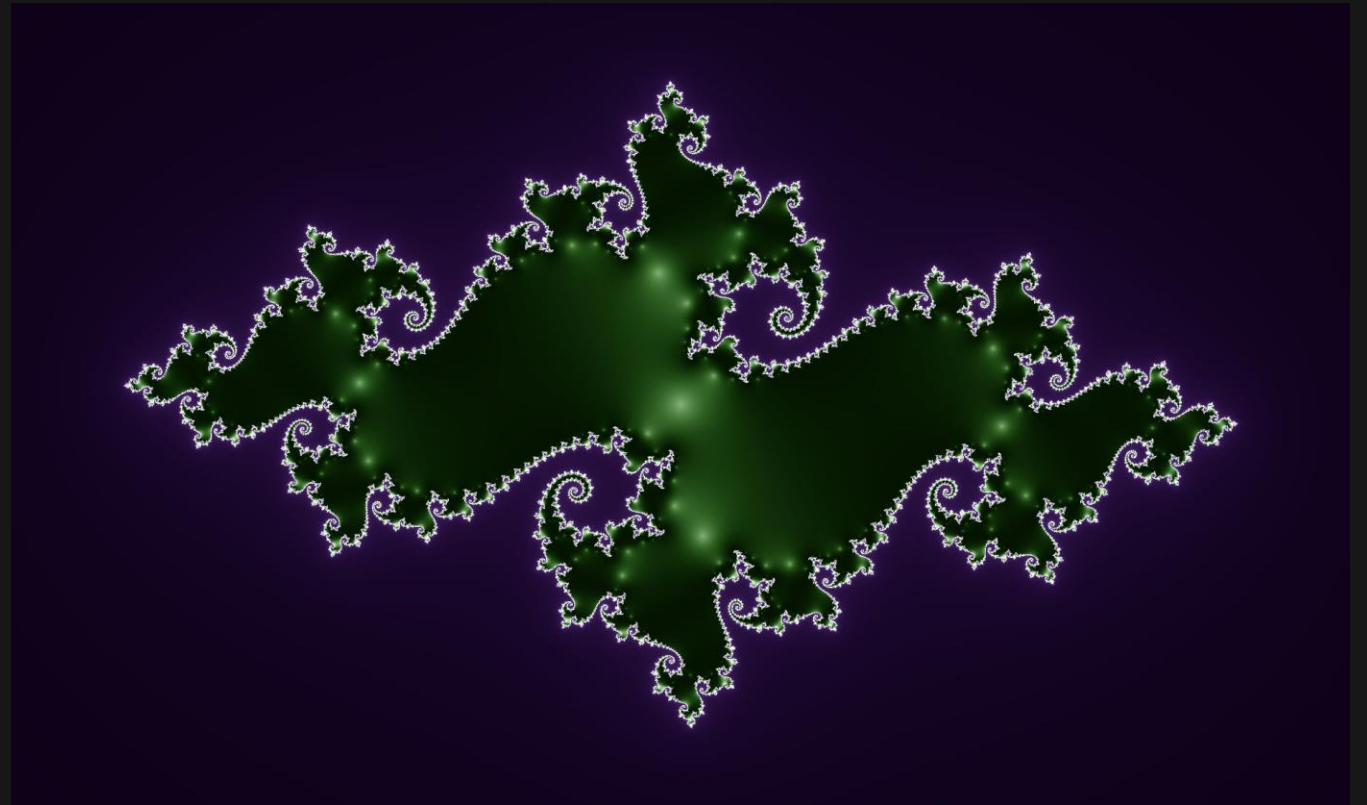
Individuelles Projekt

- Ihr werdet Zeit kriegen, um 1-n Sketches zu entwerfen
- Das individuelle Projekt sollte auf dem Gelernten aufbauen und dies erweitern, oder ein neuer Algorithmus sein
- Fügt Interaktion hinzu

Individual project ideas

Mandelbrot or julia set

- [Mandelbrot set](#)
- [Complex numbers](#)



<https://de.wikipedia.org/wiki/Julia-Menge>

Data visualization

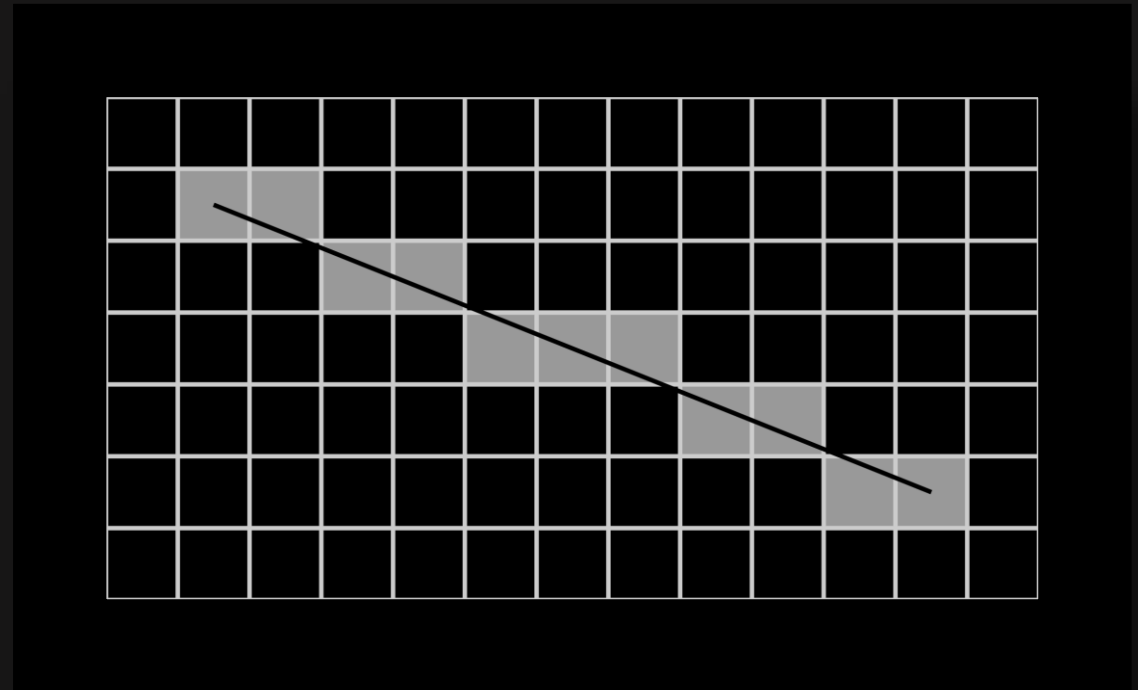
- «US Elevation Tiles»
- Potential to make it interactive
- Could be data driven
- Could point out a social problem



https://www.reddit.com/r/dataisbeautiful/comments/jslbn9/us_elevation_tiles_oc/

Bresenham's line algorithm

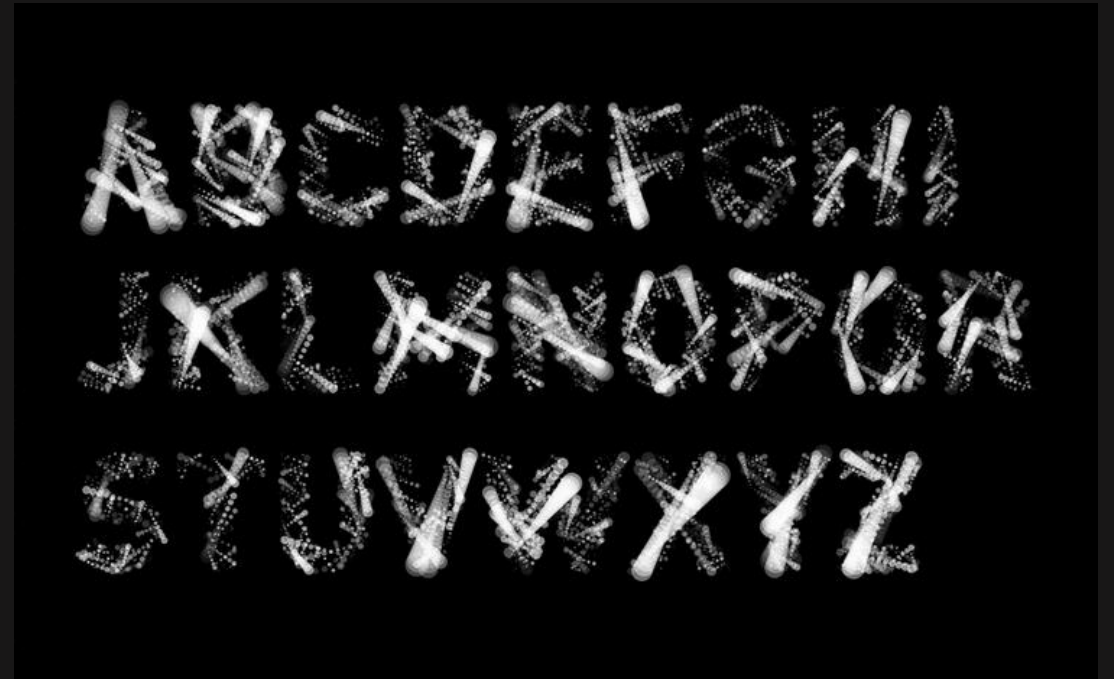
- Problem: How to draw a line?
- Extension: Midpoint circle algorithm



https://en.wikipedia.org/wiki/Bresenham%27s_line_algorithm

Algorithms with fonts

- Analysing and sorting letters of a text
- Example sketches



<https://zewang.info/Generative-Typography>

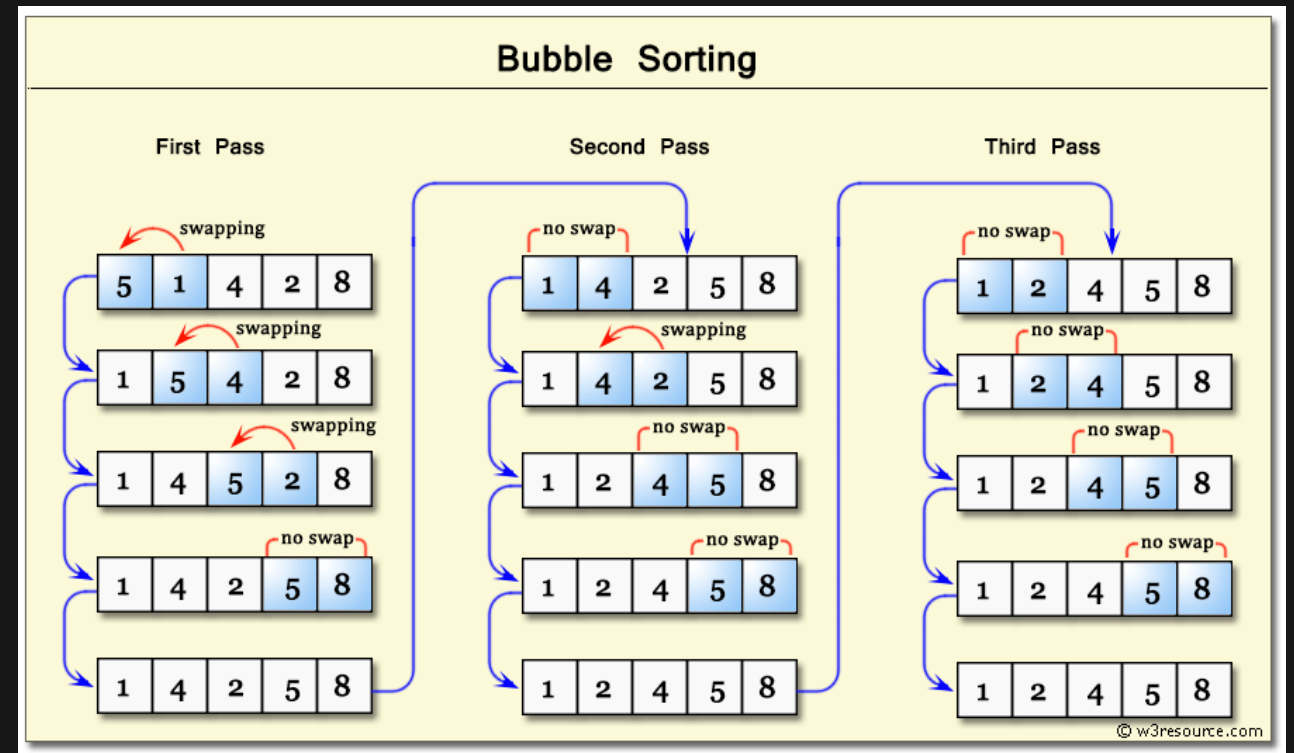
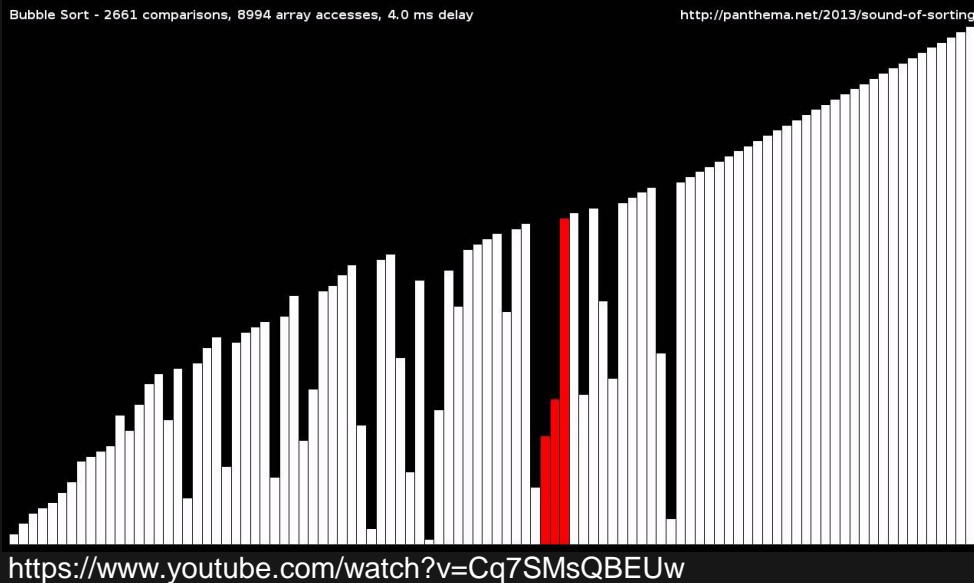
Algorithms with sound / music

- [p5.js sound library](#)
- [Example sketch](#)



Visualize sort algorithms

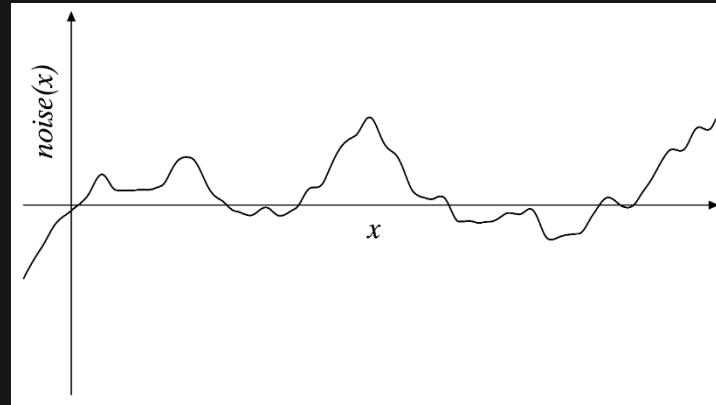
- Selection and explanation of sorting algorithms



<https://www.w3resource.com/csharp-exercises/searching-and-sorting-algorithm/searching-and-sorting-algorithm-exercise-3.php>

Noise and perlin noise

- [Explanation](#)
- [Another explanation](#)
- Show usage of perlin noise
 - In 1D, 2D, 3D



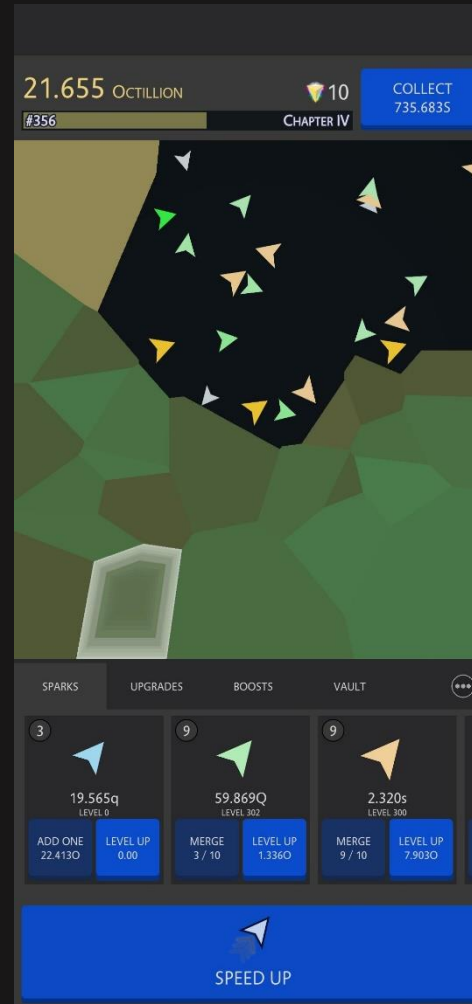
<https://blog.hirnschall.net/perlin-noise/>



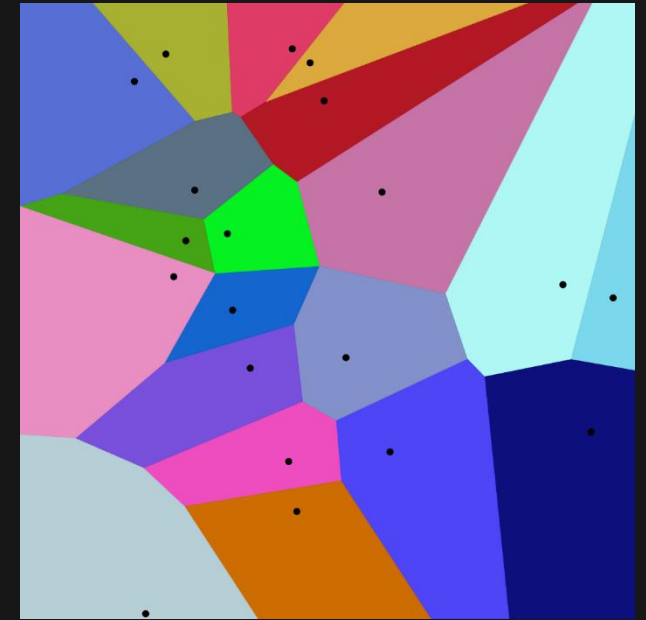
<https://rtouti.github.io/graphics/perlin-noise-algorithm>

Voronoi diagram

- Explore usage of voronoi
- [Bowyer Watson algorithm](#)
- [Fortune's sweep line algorithm](#)



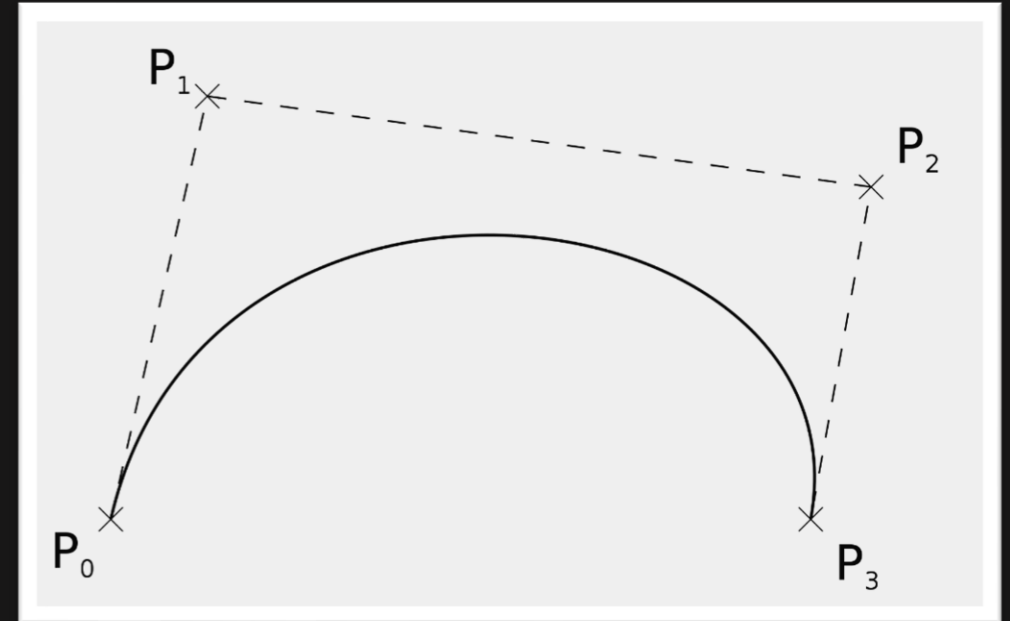
<https://www.reddit.com/r/zenshards/>



https://en.wikipedia.org/wiki/Voronoi_diagram

Bézier curve

- [p5.js bezier](#)
- Movable points (interactivity)
- CSS cubic-bezier



https://en.wikipedia.org/wiki/B%C3%A9zier_curve

Ulam spiral

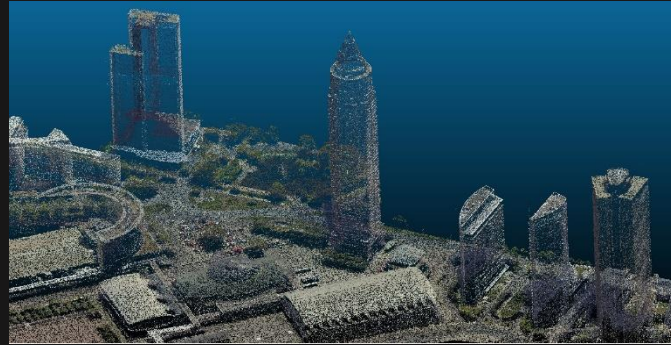
- Variants

37	36	35	34	33	32	31
38	17	16	15	14	13	30
39	18	5	4	3	12	29
40	19	6	1	2	11	28
41	20	7	8	9	10	27
42	21	22	23	24	25	26
43	44	45	46	47	48	49...

https://en.wikipedia.org/wiki/Ulam_spiral

Point clouds

- [Example sketch](#)
- X, Y, Z, R, G, B, [...]
- Check values!
 - Colors could be mapped to 0-1, or 0-255
- Animation: Lerp from pointcloud A to pointcloud B



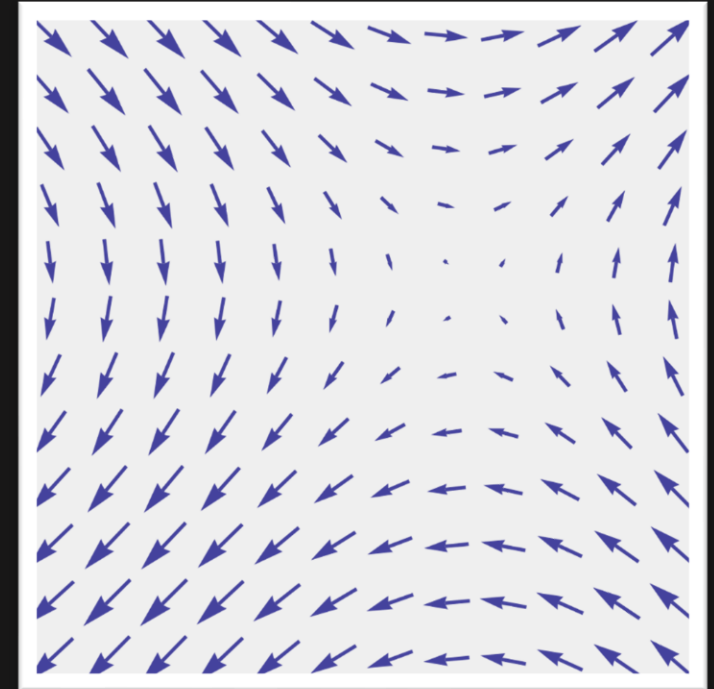
<https://docs.nframes.com/features/sure-products/3d-point-cloud/>



<https://blog.allplan.com/en/point-clouds>

Vector field

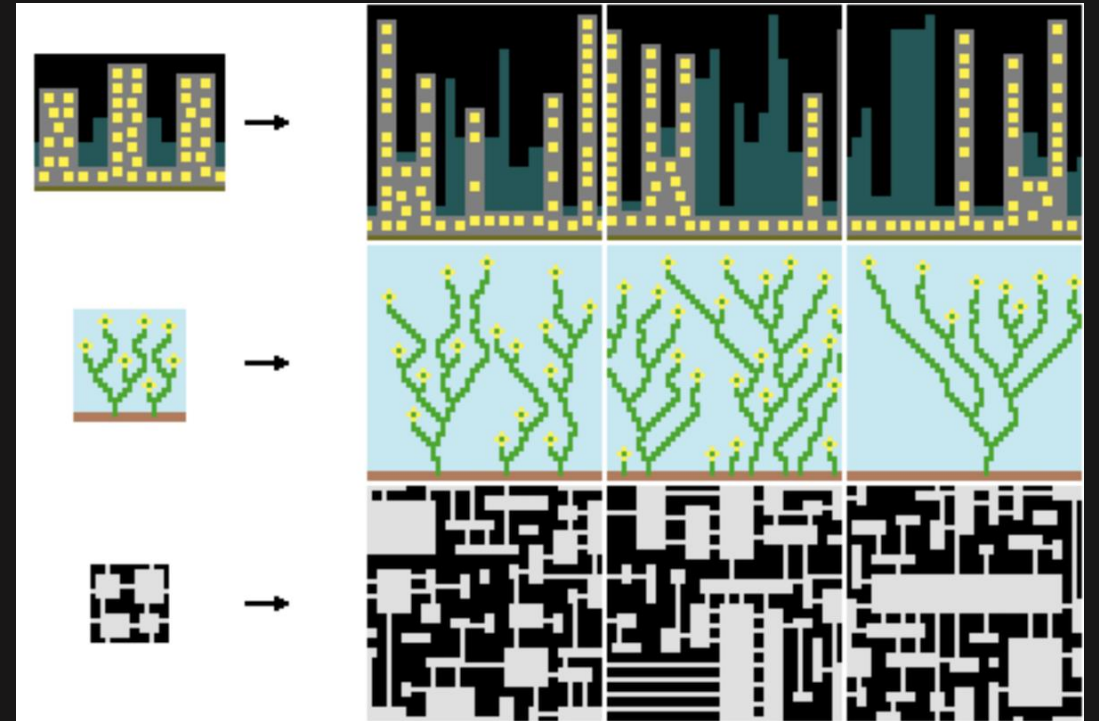
- [Example](#)
- Multiple points
- Use with fluid simulation



https://en.wikipedia.org/wiki/Vector_field

Wave collapse function

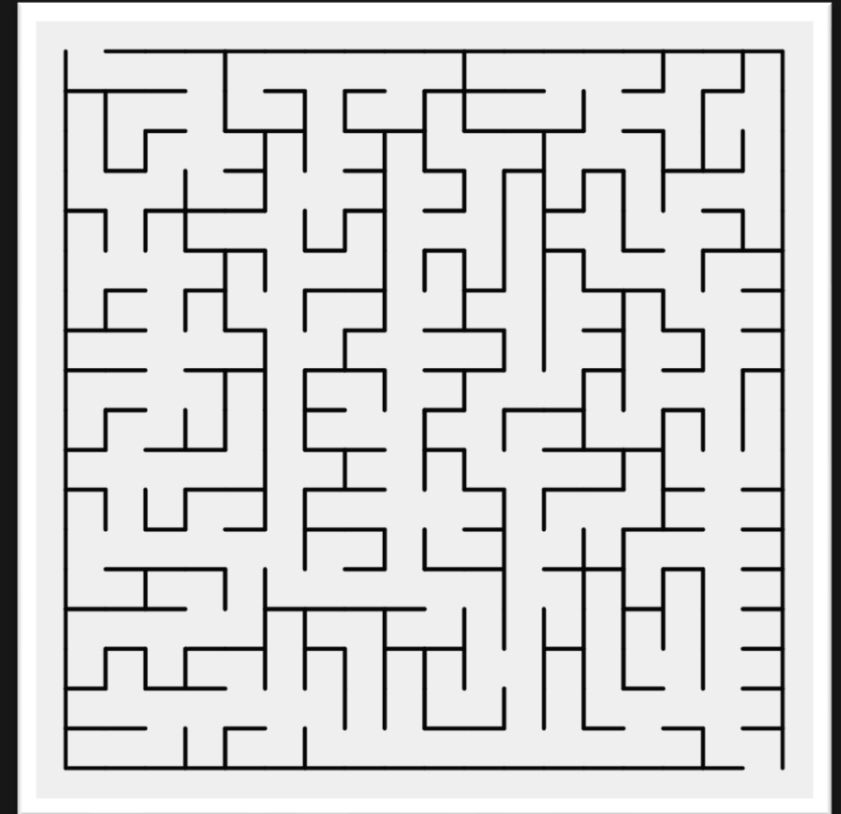
- Ascii-art
- Level generation
- Sudoku solver
- Maze generator



<https://robertheaton.com/2018/12/17/wavefunction-collapse-algorithm/>

Maze algorithm

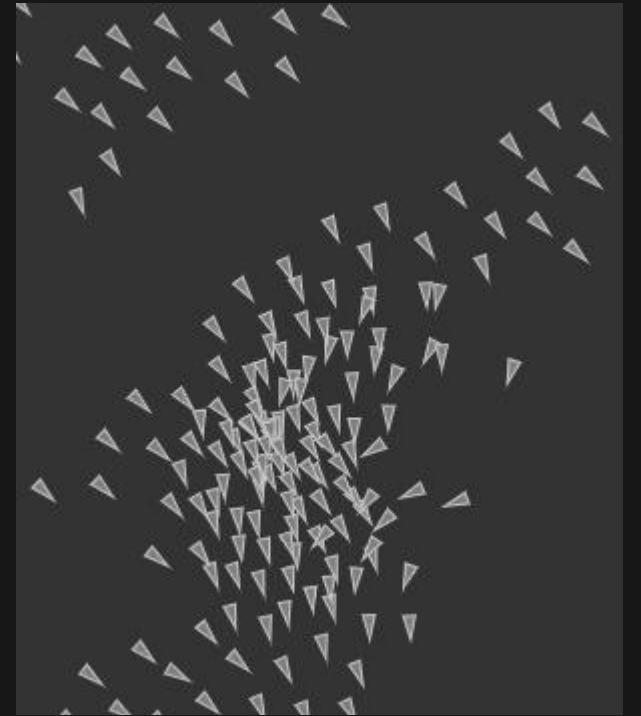
- Perfect Maze
- Imperfect maze
- Variations with different size
- Pathfinding



https://en.wikipedia.org/wiki/Maze_generation_algorithm

Swarm simulation

- [Example](#)
- Mouse interactivity
- [Explanation](#)
- [Boids algorithm](#)



<https://p5js.org/examples/simulate-flocking.html>

Kaleidoscope

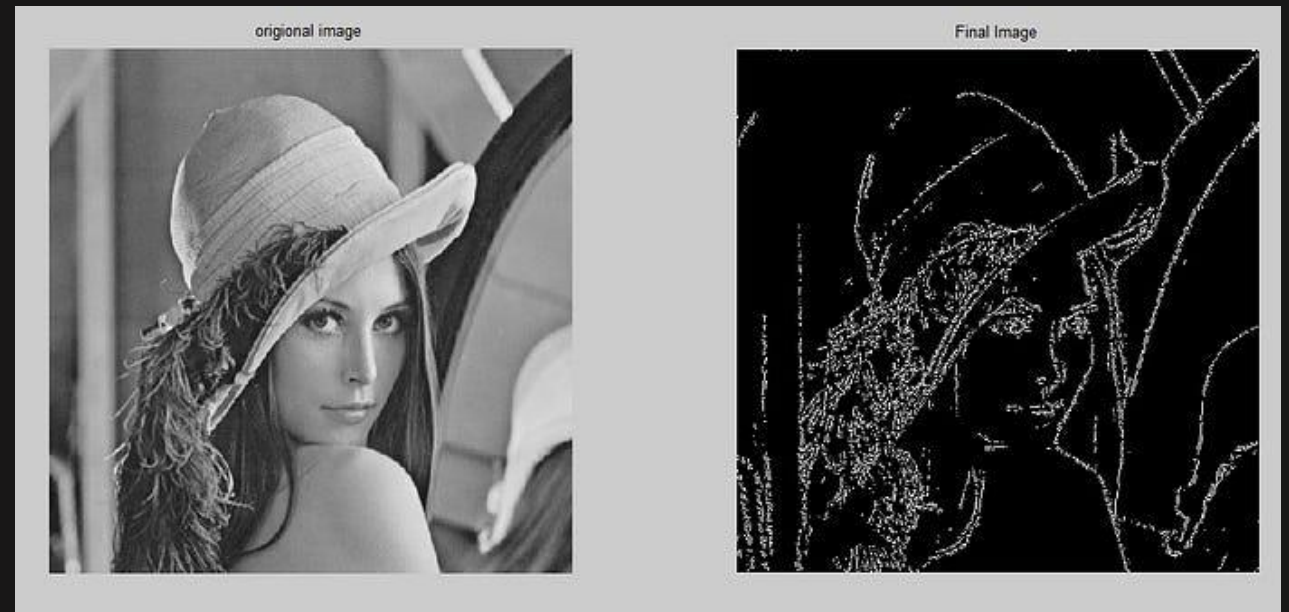
- [Example](#)
- Mouse interactivity
- Change of colors
- Define mirror axis (0-n)



<https://www.maxpixel.net/Kaleidoscope-Floral-Pattern-Glowing-Background-6955371>

Edge detection algorithm

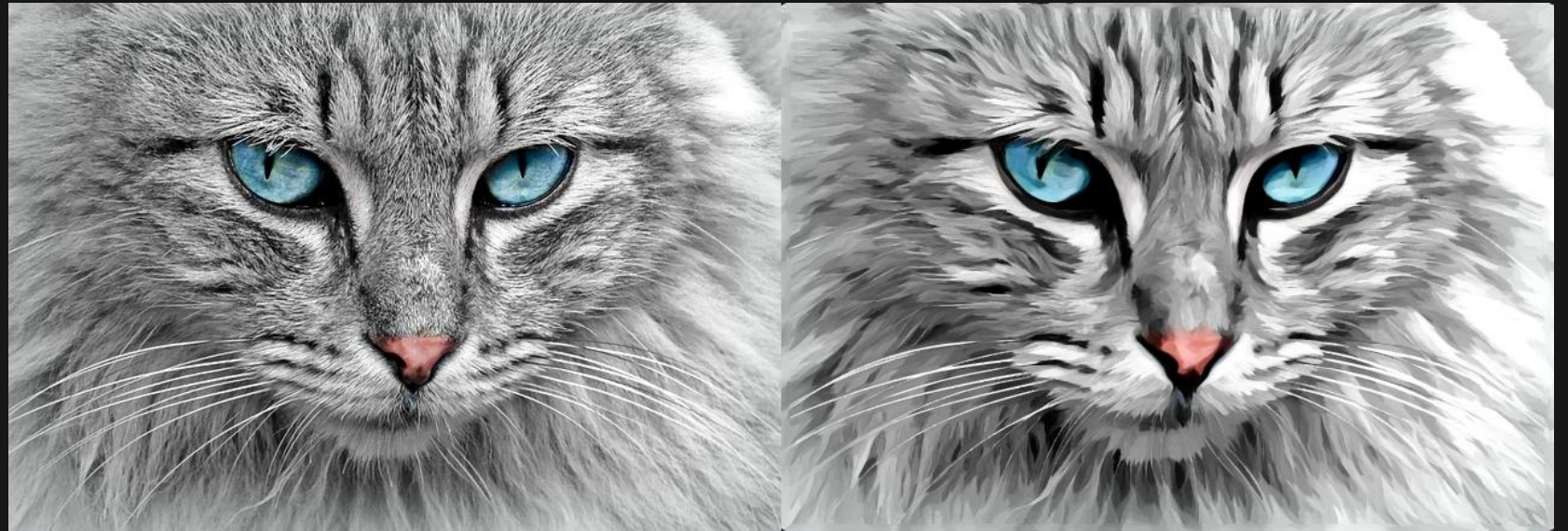
- Example sketch
 - This is not good enough



<https://medium.datadriveninvestor.com/understanding-edge-detection-sobel-operator-2aada303b900>

Kuwahara filter

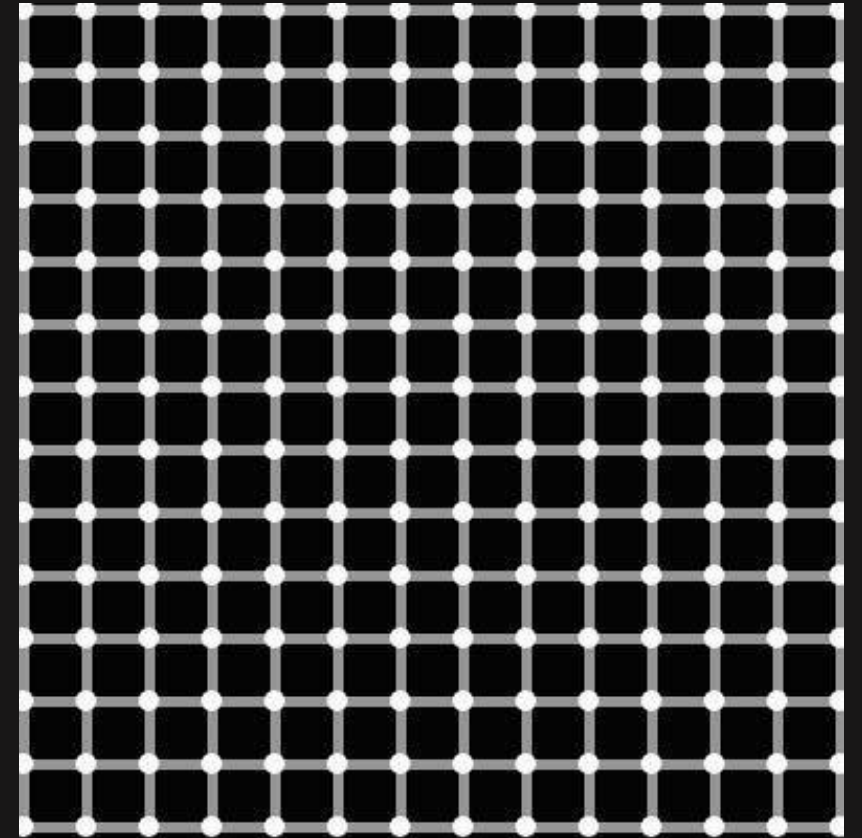
- [Explanation](#)



<https://github.com/yeataro/TD-Anisotropic-Kuwahara>

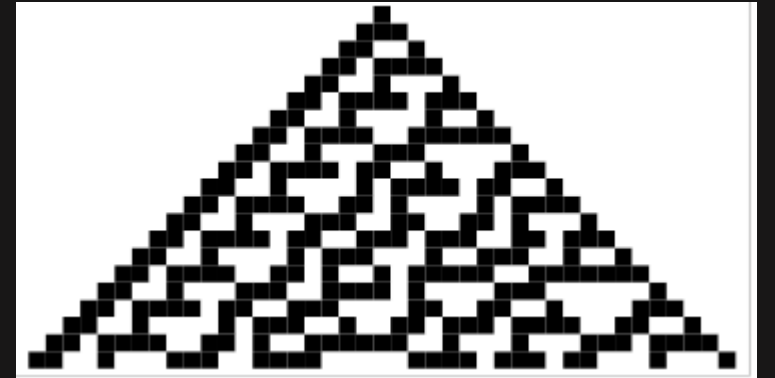
Illusions

- Look up illusions and (re)create them in p5js
- Static is ok
- Animated is preferred



Wolfram cellular automata

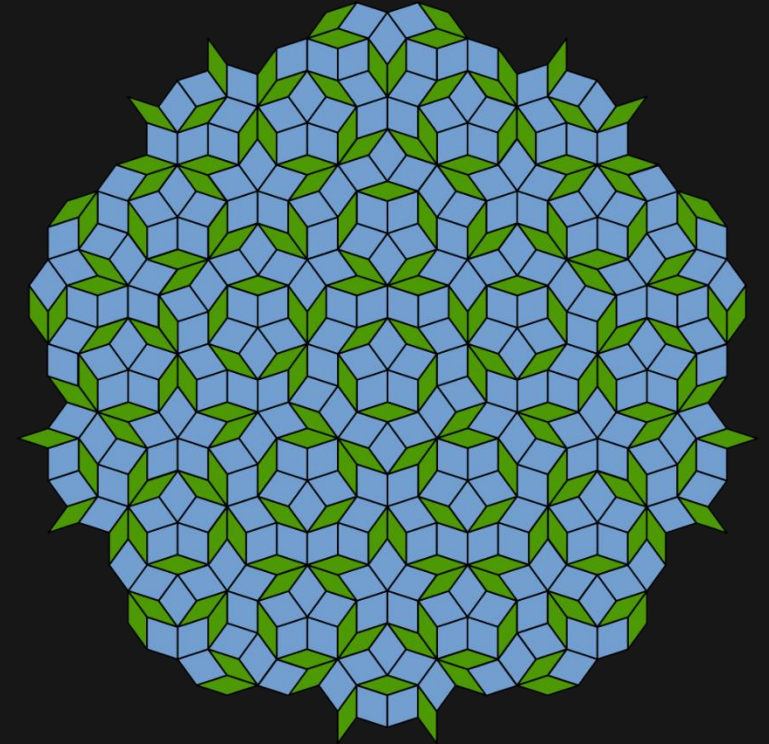
- [Rules and explanation](#)
- Make it 2D, then 3D
- Implement different rules (90, 94 etc)
- Add mouse zoom/rotation



<https://mathworld.wolfram.com/ElementaryCellularAutomaton.html>

Penrose tiles

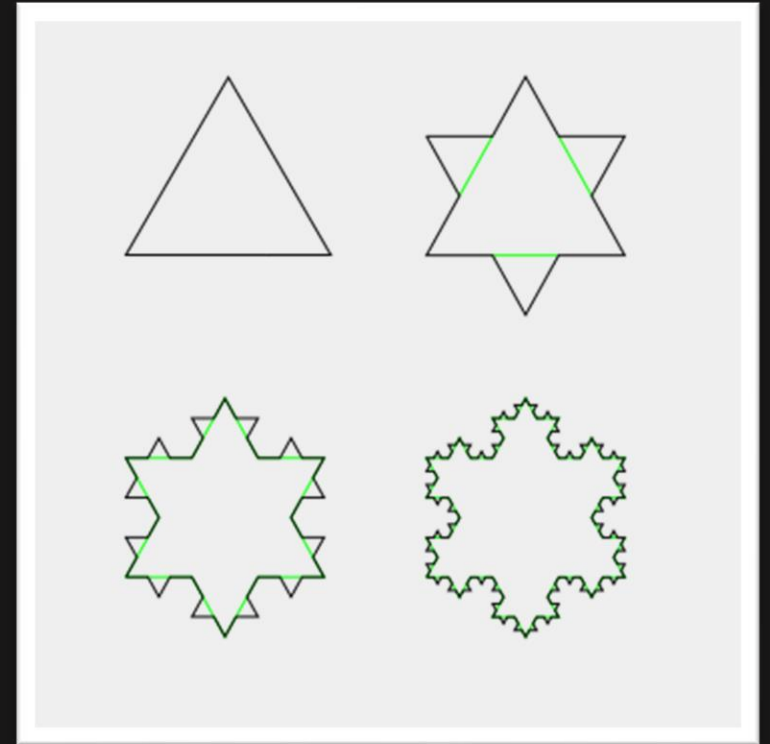
- [Example sketch](#)
- Look up the usage of penrose tiles
- How can you achieve variety?



https://en.wikipedia.org/wiki/Penrose_tiling

Koch curve

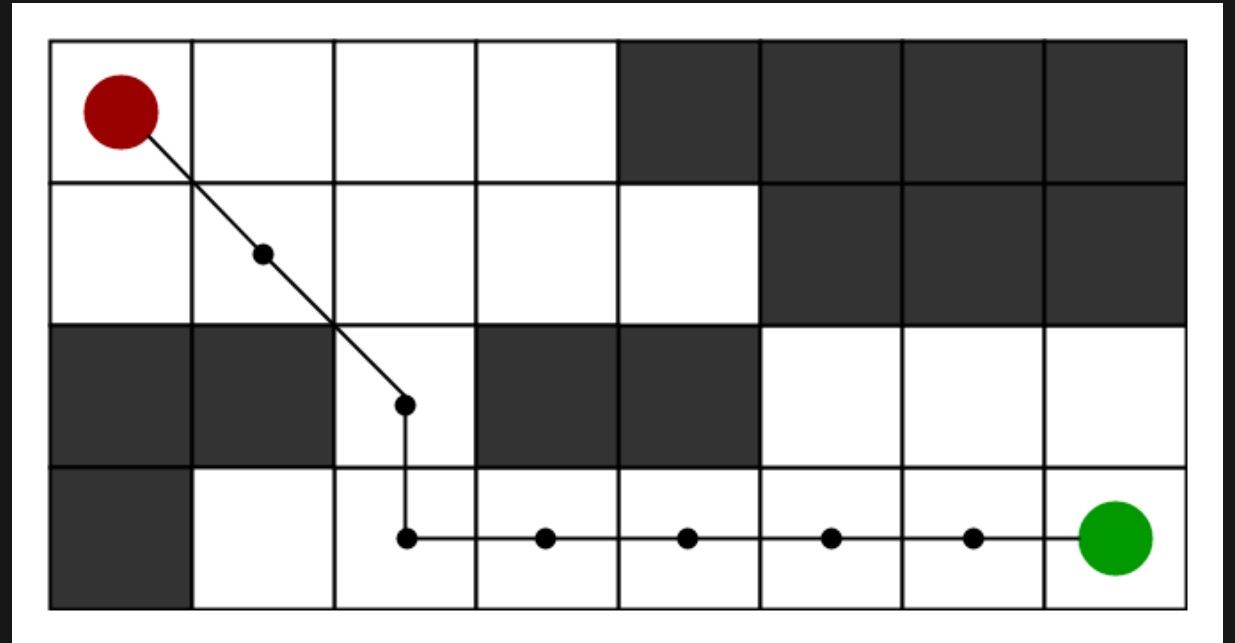
- [Example Sketch](#)
- Make it infinite zoomable
- Try the Koch
antisnowflake and other
variations



https://en.wikipedia.org/wiki/Koch_snowflake

A* pathfinding visualized

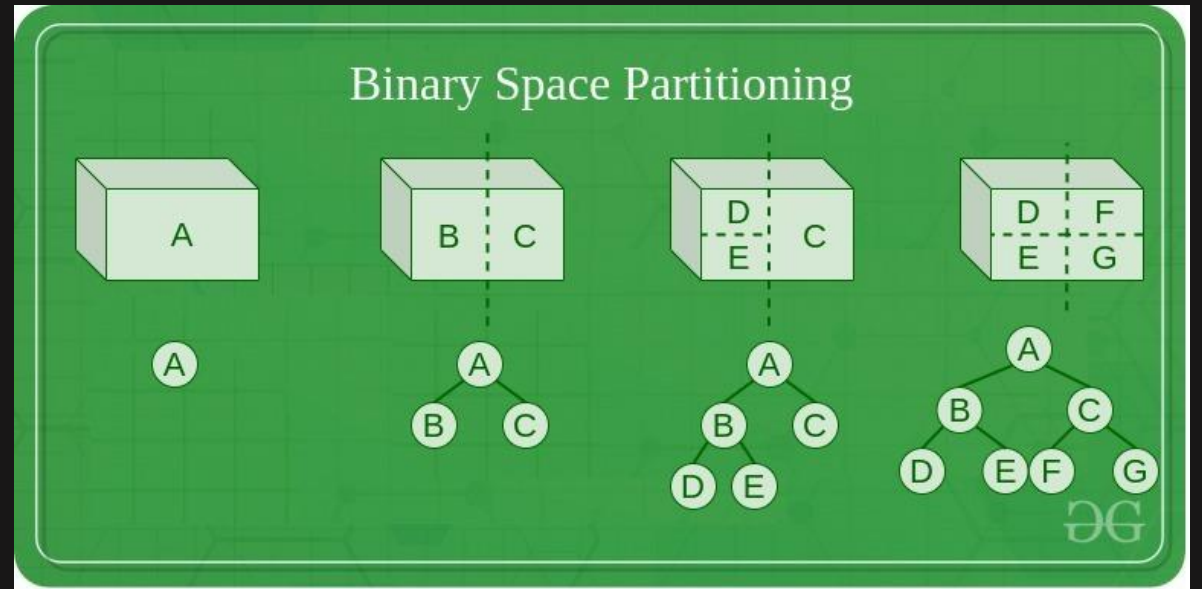
- Study Dijkstras Algorithm
- [Explanation](#)
- Visualize it!



<https://www.geeksforgeeks.org/a-search-algorithm/>

Binary space partition

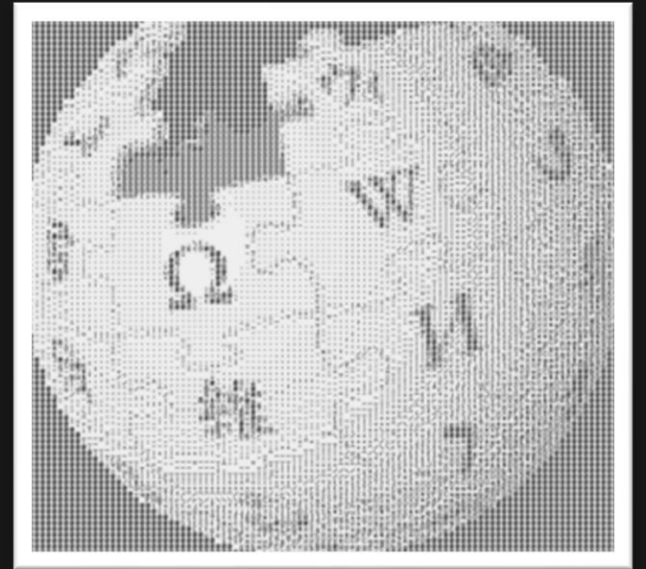
- [Explanation](#)
- [Doom and BSP](#)
- [Procedural 2D Maps](#)
- Visualize it in 2D, then 3D



<https://www.geeksforgeeks.org/binary-space-partitioning/>

Image to ascii algorithm

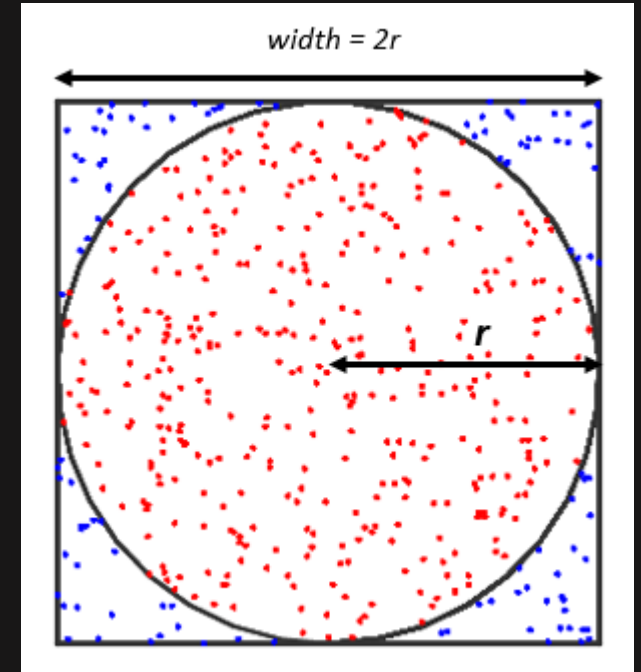
- Implement an image to ascii algorithm
- [Explanation and shadings](#)
- [Another explanation](#)
- [And another one](#)



https://en.wikipedia.org/wiki/ASCII_art

Estimate π

- Implement different variations to estimate pi
 - I.E: [Monte Carlo Method](https://www.101computing.net/estimating-pi-using-the-monte-carlo-method/)
- Visualize the steps



<https://www.101computing.net/estimating-pi-using-the-monte-carlo-method/>

And many more!

- [List of algorithms](#)
- Got own ideas? Share them!