Procedural shapes

CMSC425.01 Fall 2019

Administrivia

- Exam being graded …

Today’s question

How to create interesting shape assets

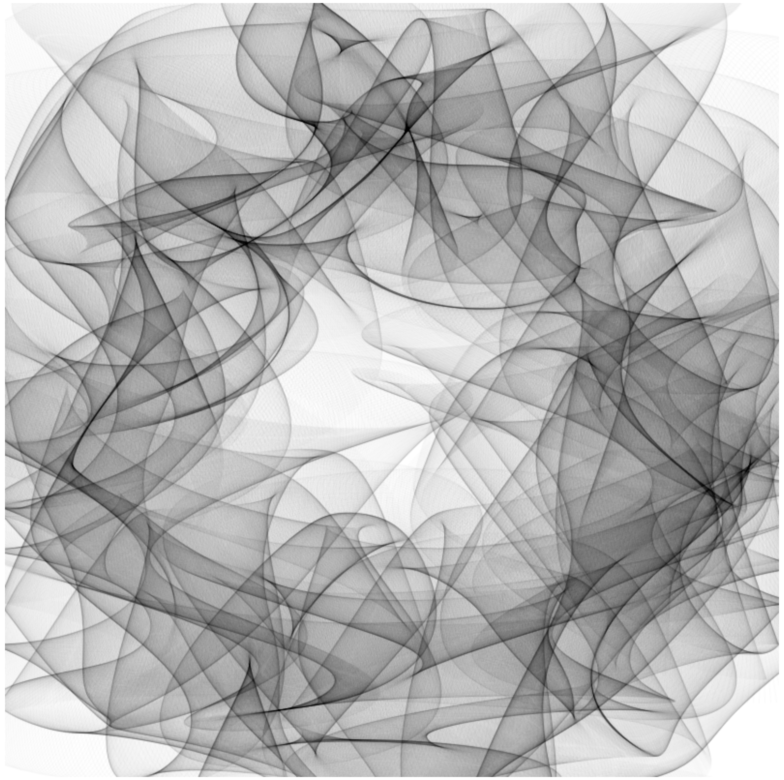
Autogenerating terrain, objects, animations

- Why
  - Efficiency – faster than by hand
  - Variety – generate variations on a template
  - LOD – Generate level of detail as needed (*billboarding* in Unity)
- How:
  - Office procedural generation: create, store mesh, add
  - Online – during game: shape represented by subroutine/object, not a fixed
- What:
  - 3D shapes, 2D textures
  - Fake real (eg, trees), non-real things

Smooth vs. articulated shapes

Bezier curves (Processing)

Continuous, smooth, parametric

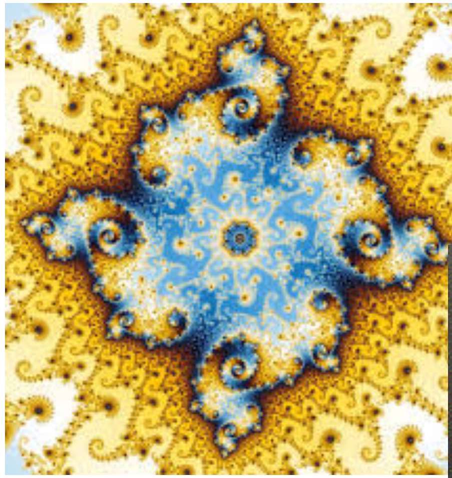
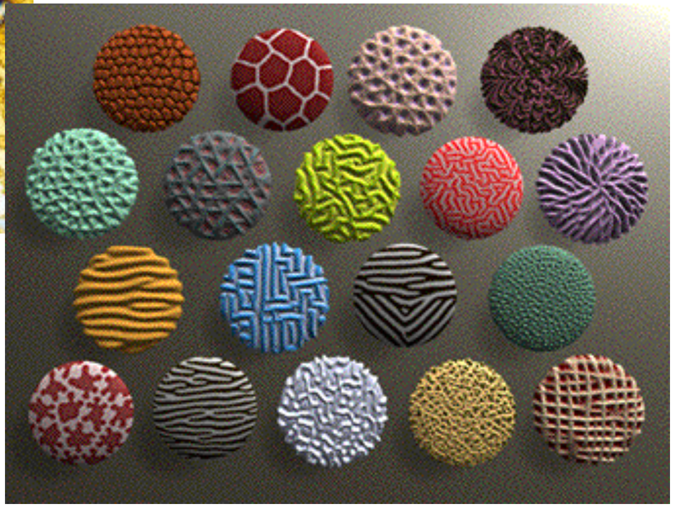
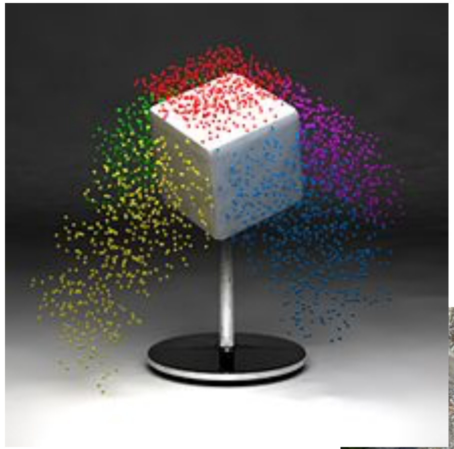


“Fractal” shapes

Articulated, subparts, discontinuous

“Fractal” examples

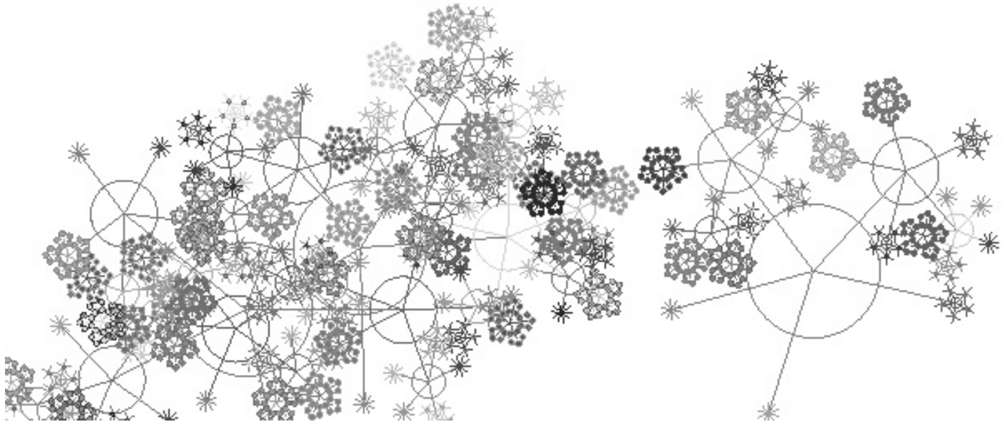
- Midpoint displacement terrain
- Particle systems
- Lindermeyer Systems (L-systems)
- Perlin noise
- Attractor sets
- Iterated function systems (IFS)
- Reaction-diffusion textures
- Recursive fractals

Lindenmayer Systems

- *The Algorithmic Beauty of Plants*
  - by P. Prunsinkiewicz and A. Lindenmayer, 2004. Chapters 1 and 2.
  - It can be accessed online from http://algorithmicbotany.org/papers/

- Grammar based approach to branching structures, including trees



shiftcontrol studio Processing

L-system

- Generative grammar + interpreter
- Use grammar to generate a string
- Use interpreter to interpret the string as a shape
- 2D and 3D versions
- http://www.kevs3d.co.uk/dev/lsystems/

L-System definition

- Grammar **G** = (*V*, ω, *P*)
- *V* Alphabet of terminals (constants) and non-terminals (variables)
- ω Start symbol or axiom
- *P* Production rules

L-System definition

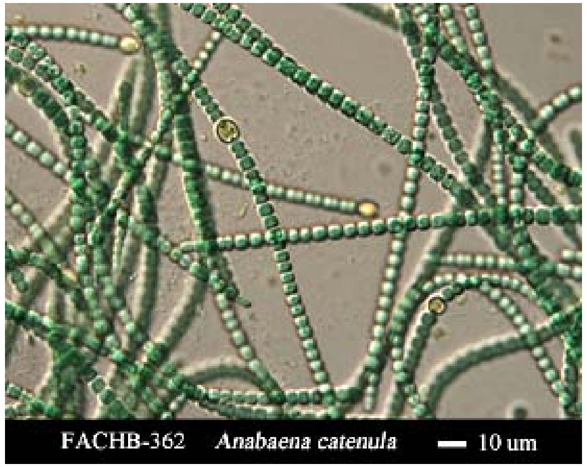
- Grammar **G** = (*V*, ω, *P*)
- *V* Alphabet of terminals (constants) and non-terminals (variables)
- ω Start symbol or axiom
- *P* Production rules

- Generative – forward derivations
- Deterministic – no choice

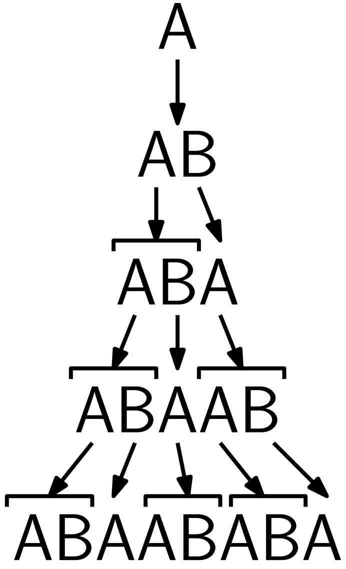
- **variables** : A B
- **constants** : none
- **axiom** : A
- **rules** : (A → AB), (B → A)

- Derivation
- A → AB → ABB → ABA → …
- ABAABABAABAABABAABABA

Anabaena catenula algae



(a)



A

AB

ABA

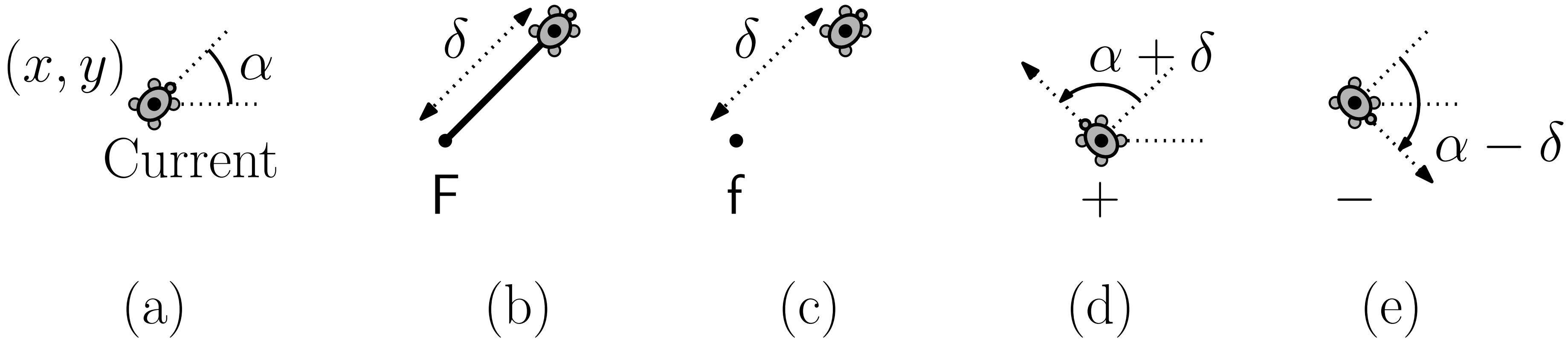
ABAAB

ABAABABA

(b)

Interpreter: Turtle graphics: state (x, y, α)

- “F”: Draw line of length d forward
- “f”: Move forward no line
- “+”: Increase the turn angle by δ
- “−”: Decrease the turn angle by δ



Turtle has: length d, turn angle δ

Example

d = 1, δ = 90

Initial state (0,0,90) (facing up)

Draw: F+F+F+F

- Shape?

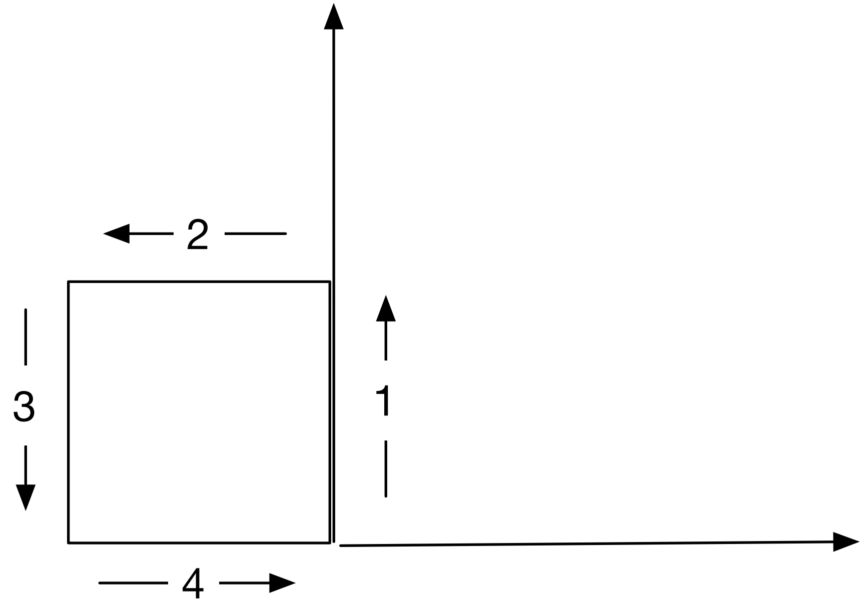
Example

d = 1, δ = 90

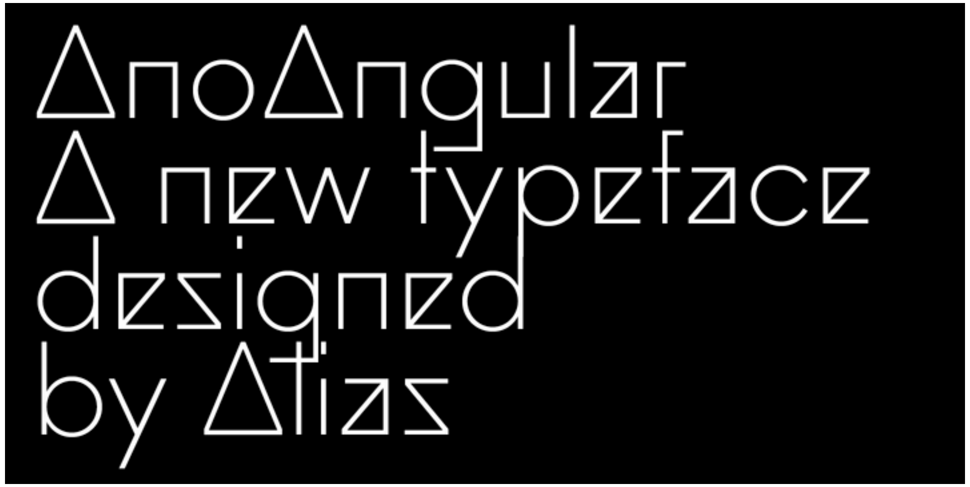
Initial state (0,0,90) (facing up)

Draw: F+F+F+F

- Shape?



Practice

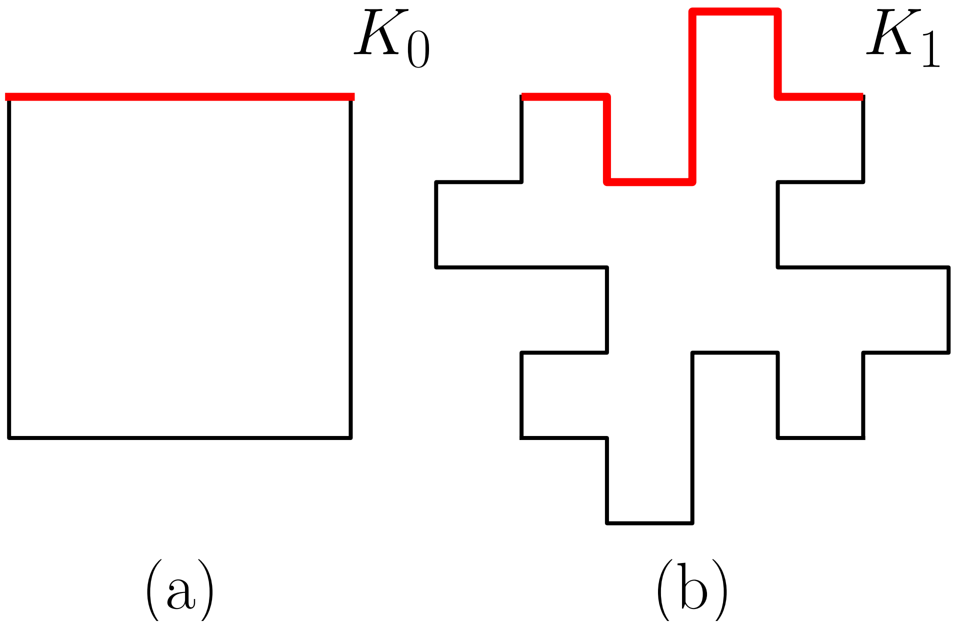


- Turtle commands and angle for

- Lower case a?
- Lower case s?
- Lower case w?
- Lower case e?

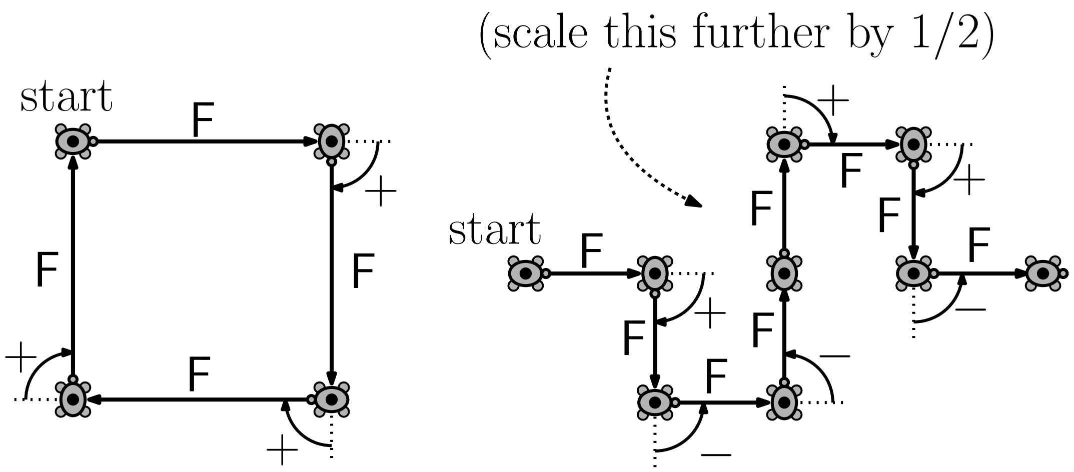
- You can start the turtle wherever you want, and at any angle. Just need to make clear.

Practice



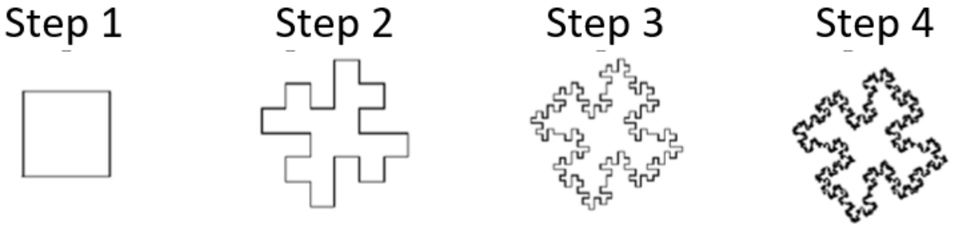
- Turtle commands for Koch island
- Axiom ω = F+F+F+F
- d = 1
- δ = ??
- F → ?

Practice



(scale this further by 1/2)

(c) (d)

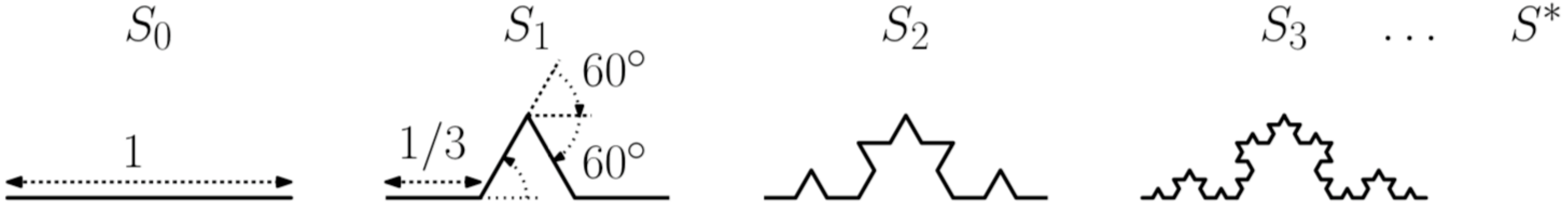


- Turtle commands for Koch island
- Axiom ω = F+F+F+F
- d = 1
- δ = ??
- F → F+F−F−FF+F+F−F

- Note: Scaling down with each generation to keep same size

In-Class Exercise 1

Consider the series of shapes $S_0, S_1, \ldots$ shown in the figure below With each step, each line segment of length x is replaced by four segments each of length $x/3$. Let $S^* = \lim_{i\to\infty} S_i$.



Derive an L-system to generate this shape. (For the sake of consistency, please start at the left endpoint.) Be sure to include the step size, angle increment (in degrees), and the initial angle. Explain your symbols and their meanings in the drawing process. Give the start string and all the production rule(s). **Hint:** All angles are multiples of 60°.

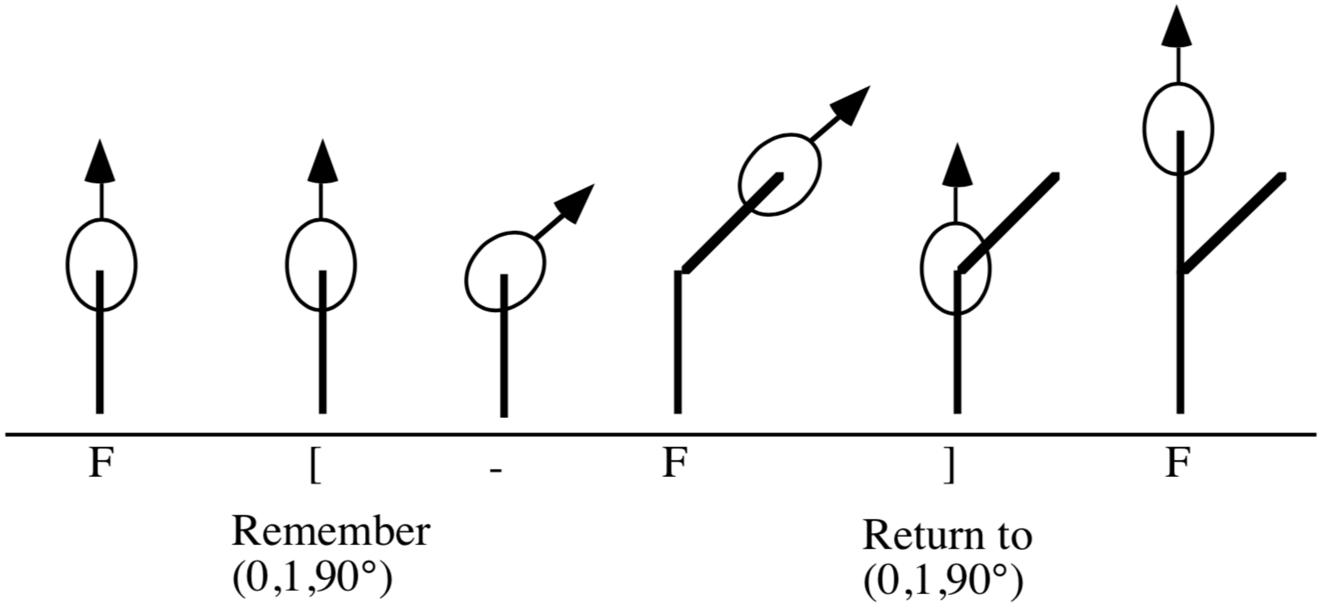
Turtle stack

- No …

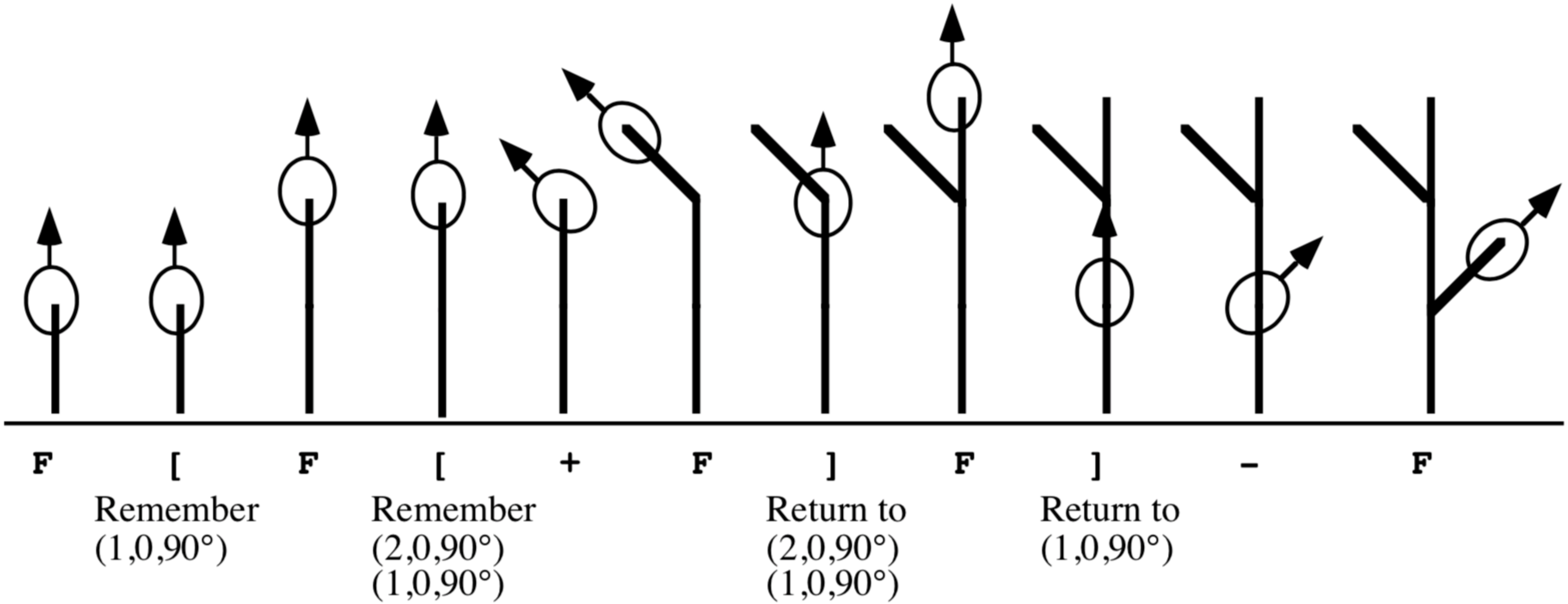
Turtle stack

- Add to vocabulary *V* :
- "[" Push state of turtle
- "]" Pop state of turtle

- Example: **F[-F]F**
- Angle 45, initial 90

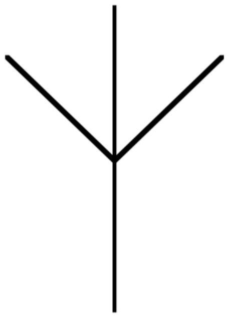


F[F[+F]F]-F

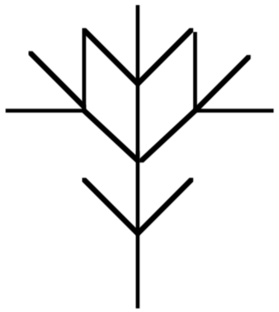

F
[
Remember
(1,0,90°)
F
[
Remember
(2,0,90°)
(1,0,90°)
+
F
]
Return to
(2,0,90°)
(1,0,90°)
F
]
Return to
(1,0,90°)
–
F

Symmetric bush 1

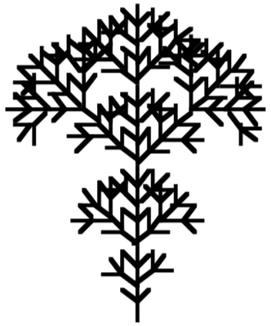
- V = { F, + , -, [,] }
- ω = F
- δ = 45
- F → F[+F]F[-F]F



N=1



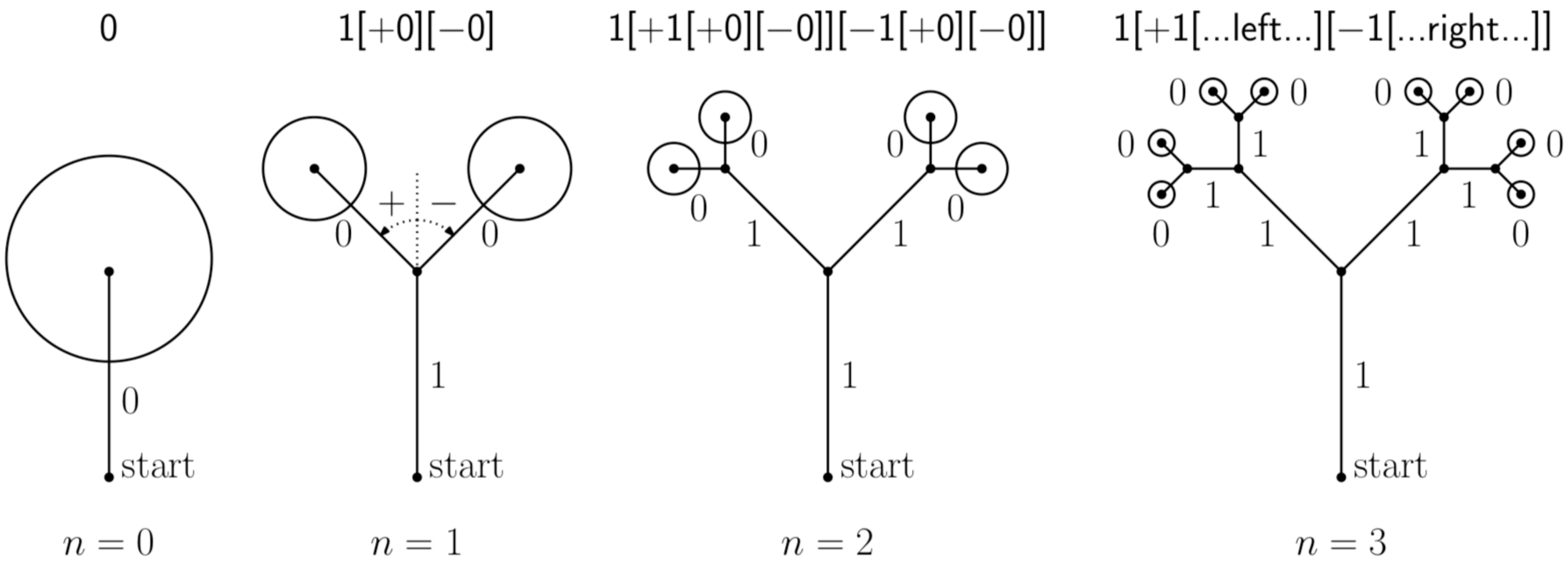
N=2



N=3

Symmetric bush 2

- V = {0,1,+,−,[,]}
- ω=0
- δ = 45
- P = {0→1[+0][−0]}

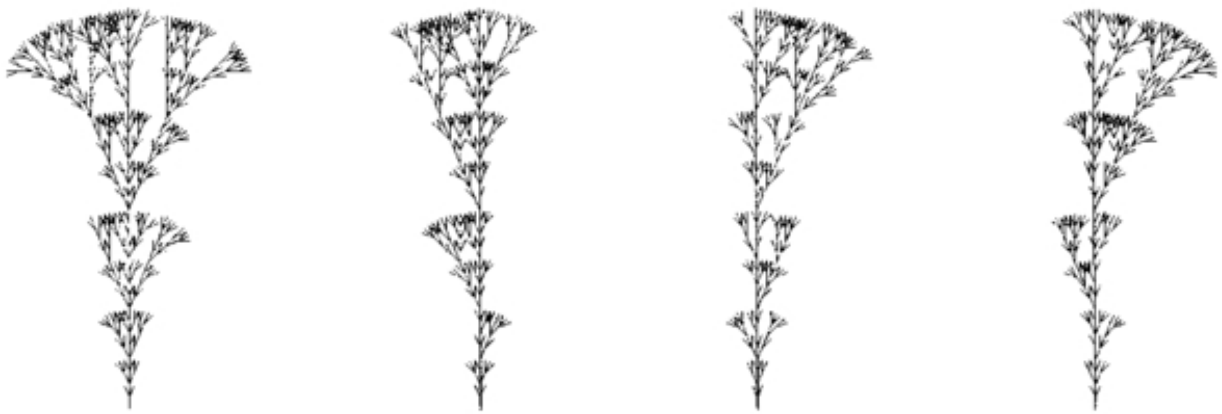


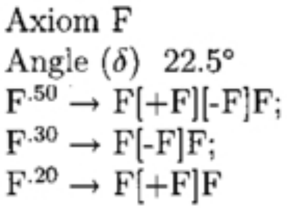
$$\begin{aligned}
n = 0 \quad &: \quad 0 \\
n = 1 \quad &: \quad 1[+0][-0] \\
n = 2 \quad &: \quad 1[+1[+0][-0]][-1[+0][-0]] \\
n = 3 \quad &: \quad 1[+1[+1[+0][-0]][-1[+0][-0]]][-1[+1[+0][-0]][-1[+0][-0]]]
\end{aligned}$$

Stochastic L-systems: Randomize

- Randomize production selection
- Choose rule 1 with p=0.4, rule 2 with p=0.6
- $a \rightarrow_{[0.4]} a\ [b]$
- $a \rightarrow_{[0.6]} b\ [a]\ b$

- Randomize parametric vocabulary
- F -> F(d) +(δ) F(d’)





Axiom F

Angle (δ) 22.5°

$F^{.50} \rightarrow F[+F][-F]F;$

$F^{.30} \rightarrow F[-F]F;$

$F^{.20} \rightarrow F[+F]F$

3D L-system

- Add to vocabulary 3D turns

$$\mathbf{R_U}(\alpha) = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

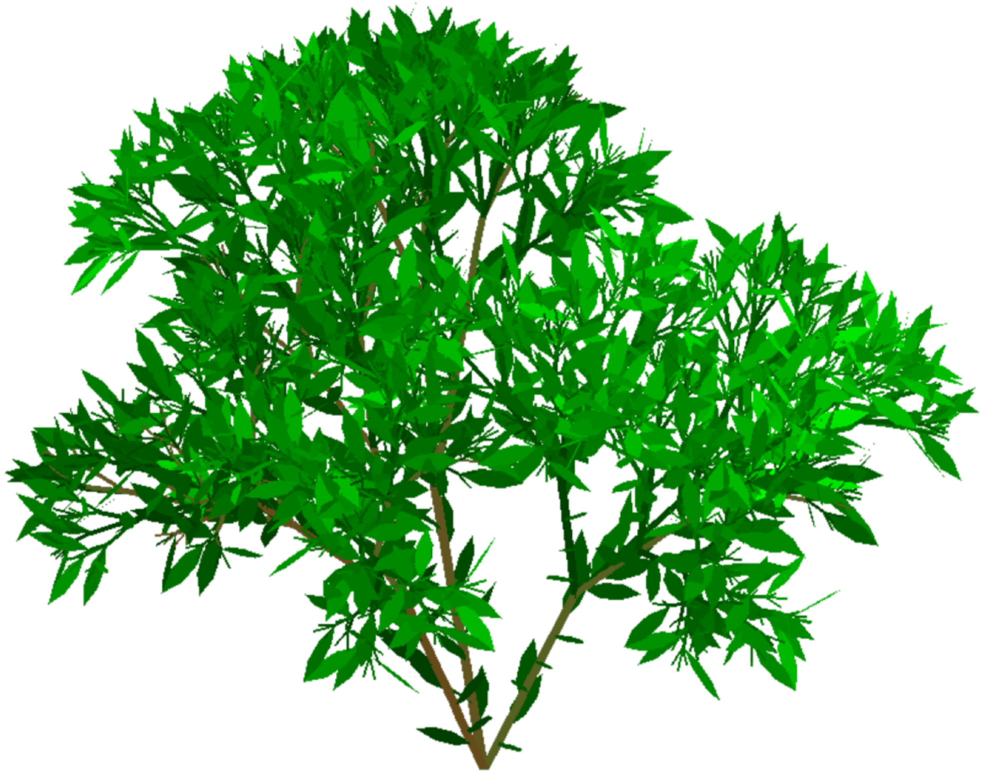
$$\mathbf{R_L}(\alpha) = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix}$$

$$\mathbf{R_H}(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

- + Turn left by angle δ, using rotation matrix RU(δ).
- − Turn right by angle δ, using rotation matrix RU(−δ).
- & Pitch down by angle δ, using rotation matrix RL(δ).
- ∧ Pitch up by angle δ, using rotation matrix RL(−δ).
- \ Roll left by angle δ, using rotation matrix RH(δ).
- / Roll right by angle δ, using rotation matrix RH(−δ).
- | Turn around, using rotation matrix RU (180◦).

n=7, δ=22.5°

ω : A
p_1 : A → [&FL!A]/////'[&FL!A]///////'[&FL!A]
p_2 : F → S ///// F
p_3 : S → F L
p_4 : L → ['''∧∧{-f+f+f-|-f+f+f}]



Making ”real” trees

- Model real trees
- Manage self-intersections
- Add animation features
  - Sway, bend correctly
  - Grow from sapling

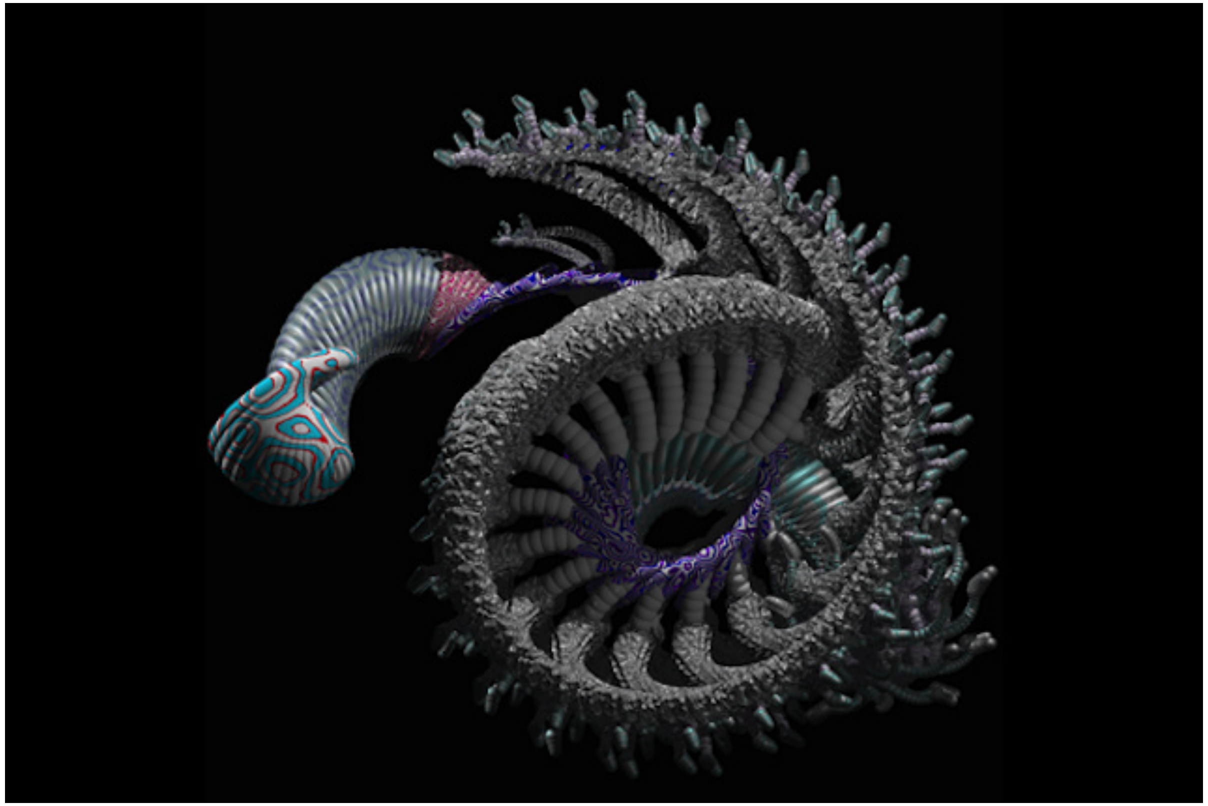
The Grove release 7

Opens up to the world with exciting new possibilities...

Not just trees

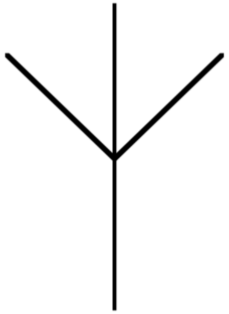
- Gian Carlo Mingati

- William Latham/Stephen Todd

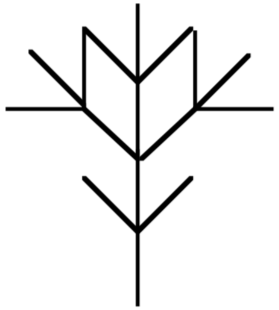


Self-similarity and fractal dimension

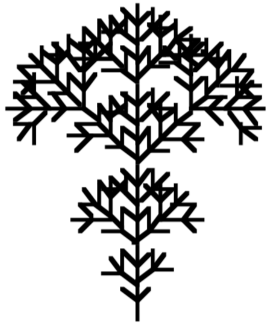
- L-system shapes are *self-similar*
- Part looks like whole



N=1

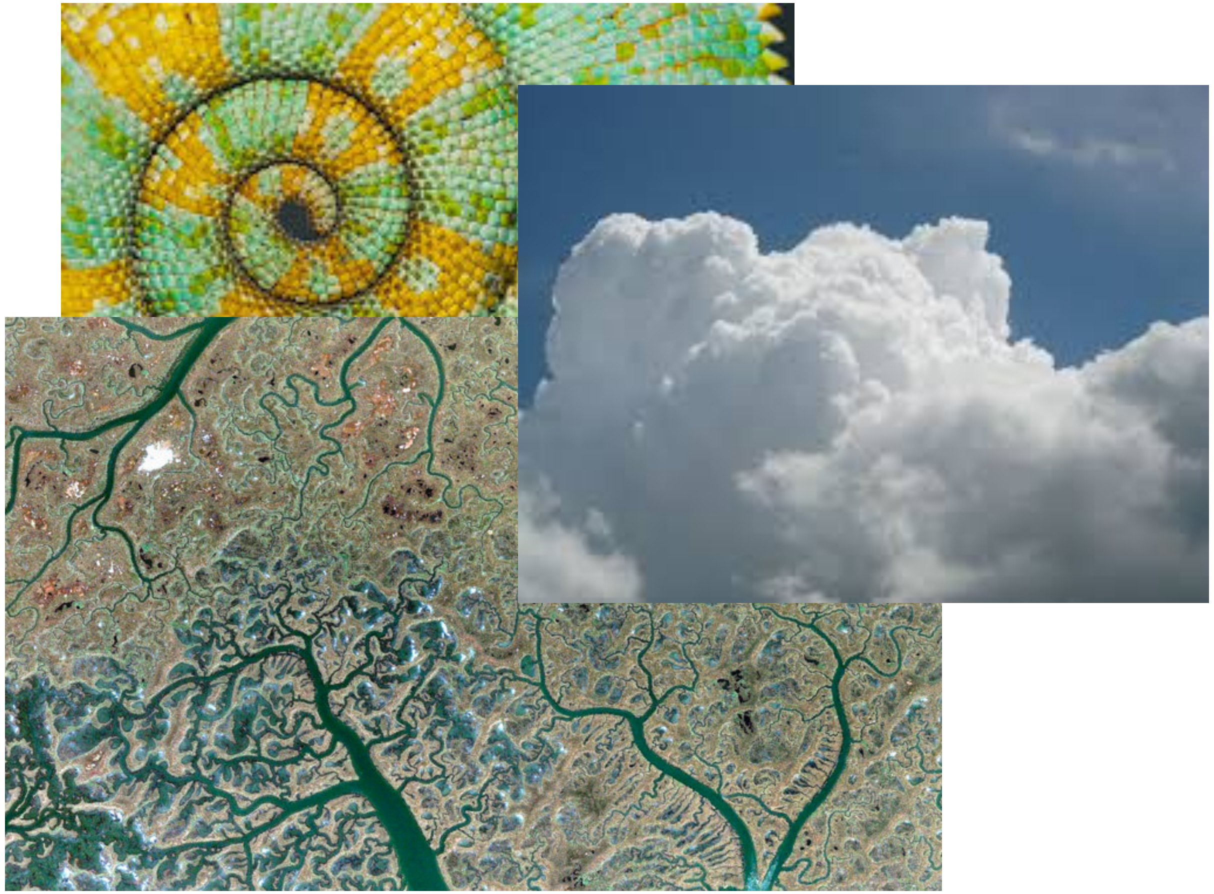


N=2



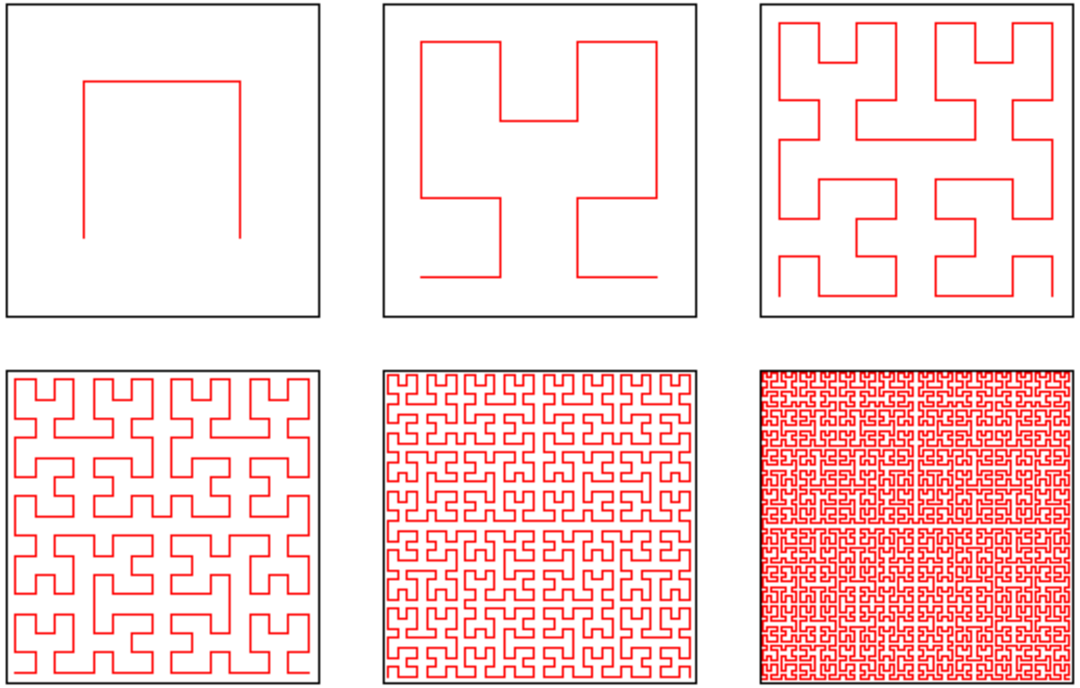
N=3

- Typical of many natural objects



Space filling curves

- Hilbert curves



Dimension

- What is the dimension of a set of points?

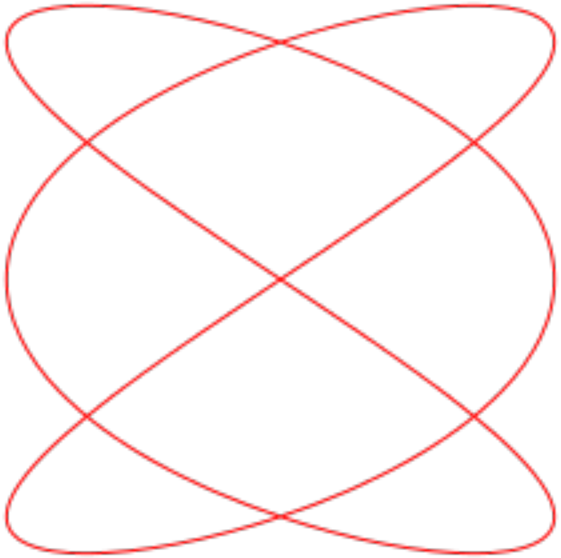
- Point – zero dimensions
- Line – 1 dimension
- Plane – 2 dimensions
- Box – 3 dimensions

Dimension

- What is the dimension of a set of points?

- Point – zero dimensions
- Line – 1 dimension
- Plane – 2 dimensions
- Box – 3 dimensions

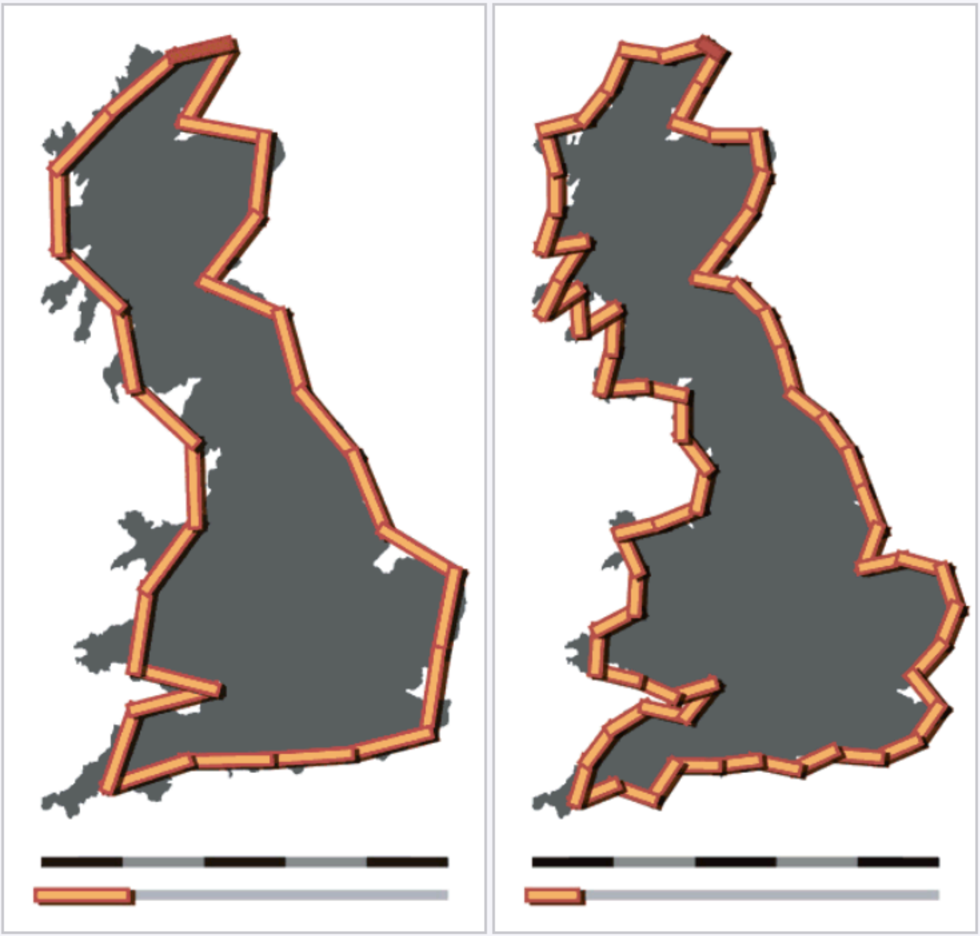
- Curve?
  - Lives in 2D
  - Only takes 1D to trace out



Fractal dimension

- Measure of complexity of curve or surface
- How many parameters needed to describe?
- Related to self-similarity

- *Hausdorff* measure

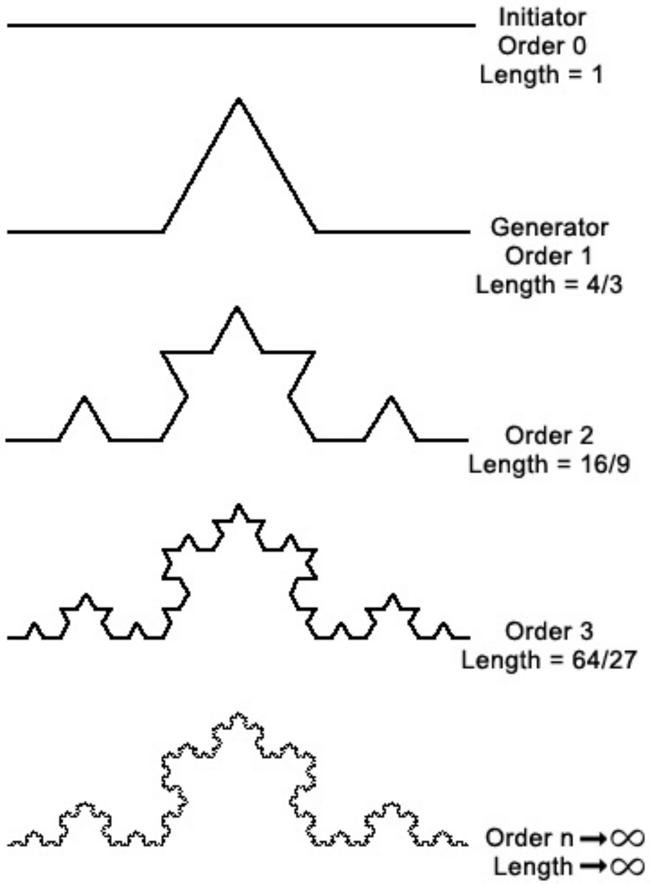


An example of the coastline paradox. If the coastline of Great Britain is measured using units 100 km (62 mi) long, then the length of the coastline is approximately 2,800 km (1,700 mi). With 50 km (31 mi) units, the total length is approximately 3,400 km (2,100 mi), approximately 600 km (370 mi) longer.

Length of fractal curve as $n \to \infty$

- Koch curve
- Each generation
  - 4 new parts
  - Each length 1/3
  - New length 4/3 * old

- $\lim_{n \to \infty} \left(\frac{4}{3}\right)^n = \infty$

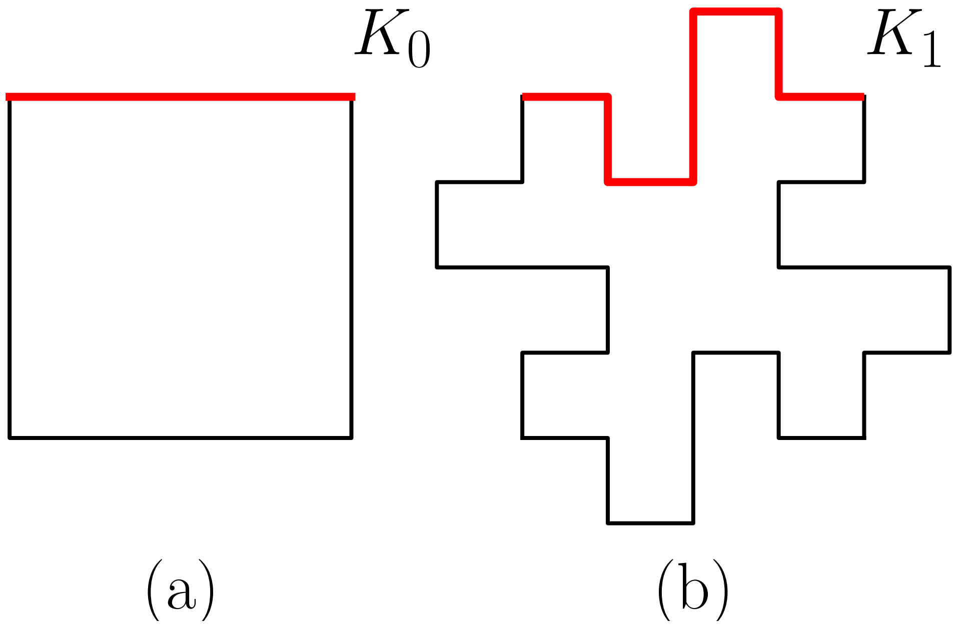


Koch island fractal dimension

- b = # of parts in recursive rule
- a = scale factor

- Here

  b = 8
  s = ¼ = 1/a so a = 4



Koch island fractal dimension

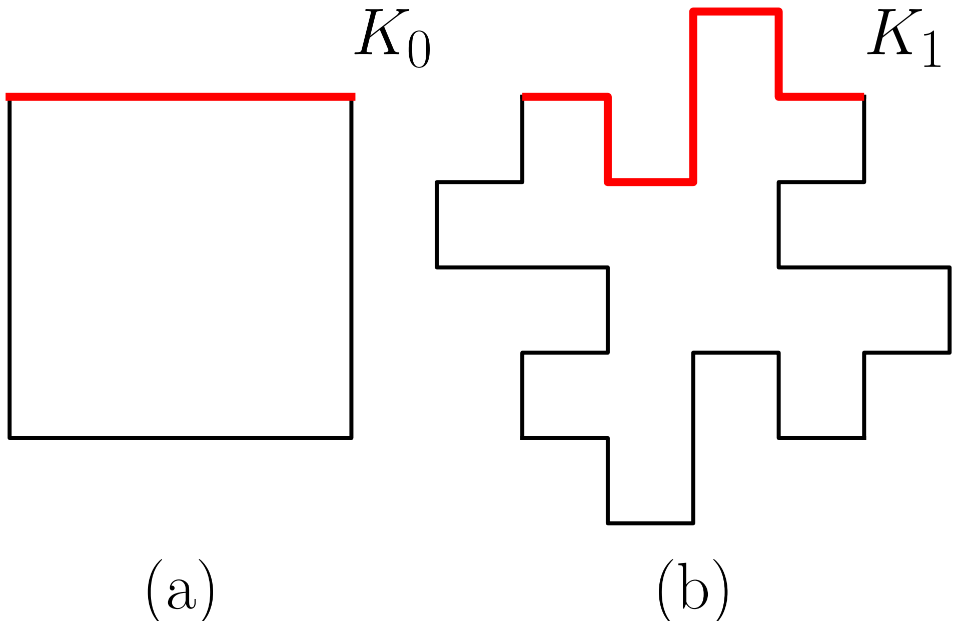
- b = # of parts in recursive rule
- a = scale factor

- Here

  b = 8

  s = ¼ = 1/a so a = 4

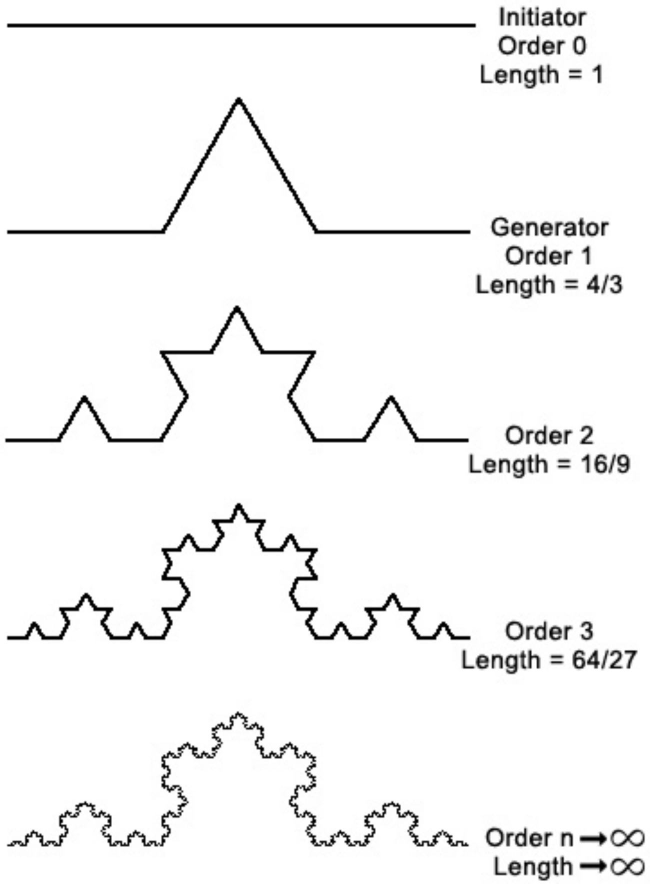
- $D = \frac{\log(b)}{\log(a)} = \frac{\log(8)}{\log(4)} = \frac{3}{2} = 1.5$



Koch curve dimension?

- b = ??
- a = ??

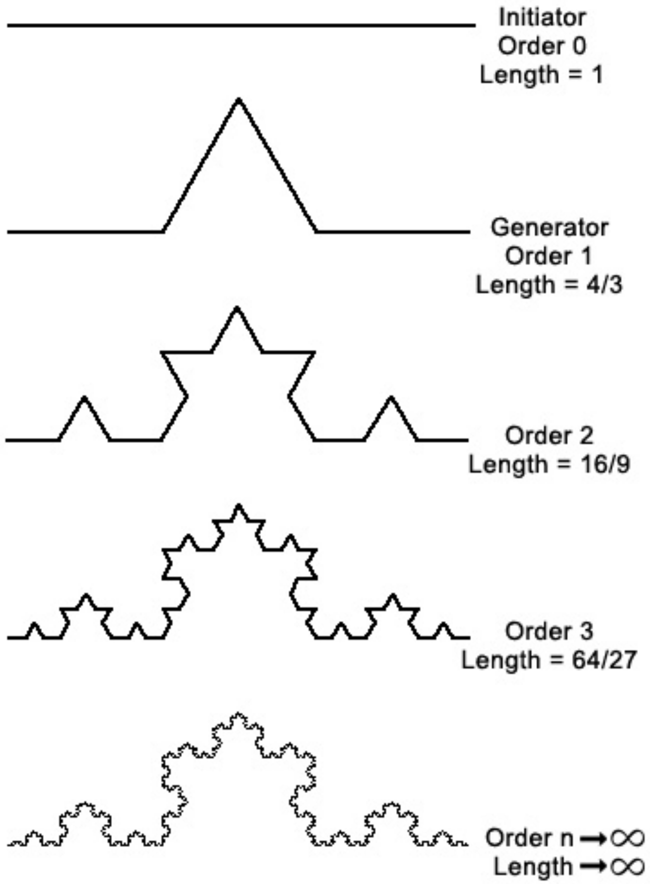
- $D = \dfrac{\log(b)}{\log(a)} = ??$



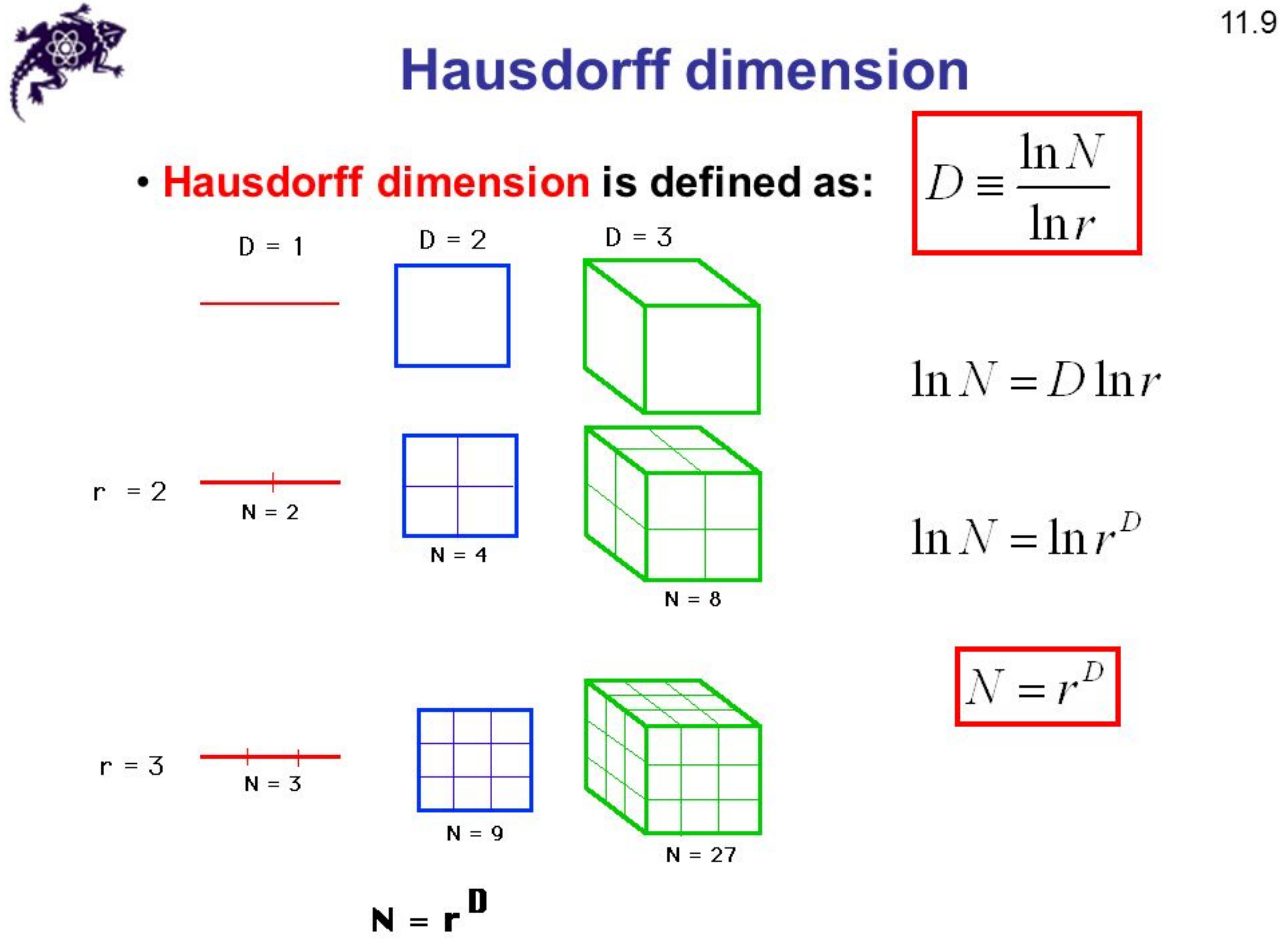
Koch curve dimension?

- b = 4
- a = 3

- $D = \frac{\log(4)}{\log(3)} \approx 1.26186$



Fractal dimension of line, place, box?



11.9

Hausdorff dimension

- **Hausdorff dimension** is defined as: $\boxed{D \equiv \dfrac{\ln N}{\ln r}}$

$$\mathbf{N = r^{D}}$$

$$\ln N = D \ln r$$

$$\ln N = \ln r^{D}$$

$$\boxed{N = r^{D}}$$

Purpose for fractal dimension

- First, fun. Interest calculation
- Second, can measure fractal dimension of real objects, match in your simulation
- Third, watch out for d = 2 or 3!
- (Space filling …)