COMPUTER GRAPHICS REALISM

CONTENT

- TILING THE PLANE
- RECURSIVELY DEFINED CURVES
- KOCH CURVES
- C CURVES
- DRAGONS
- SPACE FILLING CURVES
- FRACTALS
- GRAMMAR BASED MODELS
- TURTLE GRAPHICS
- RAY TRACING

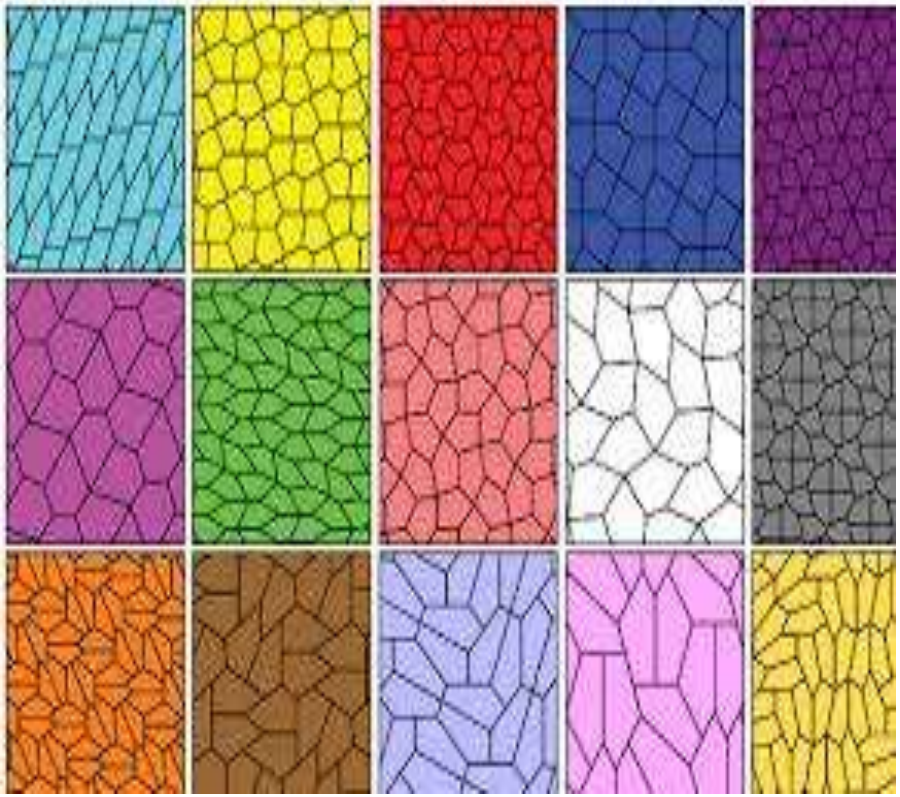
Computer Graphics Realism

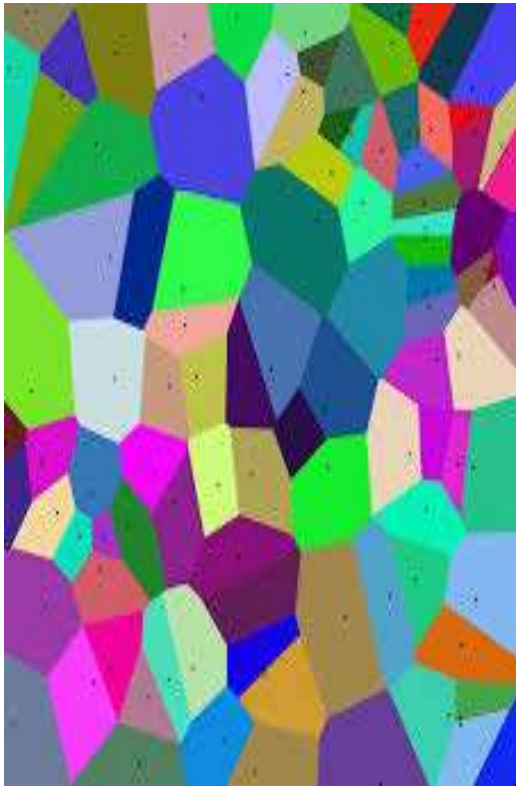
- Image is a visual representation of scene, it represent selected properties of scene to viewer with varying degree of realism.

Tiling the Plane

- Use one or more geometric shapes
- Tessellation(without gaps) of flat surface
- Shape repeated
- Moving infinity
- Covering entire plane
- Used arts,mosaics,wall papers,tiled floor

Tiling the Plane





Tiling the Plane

Tiling the Plane

Types of tiling

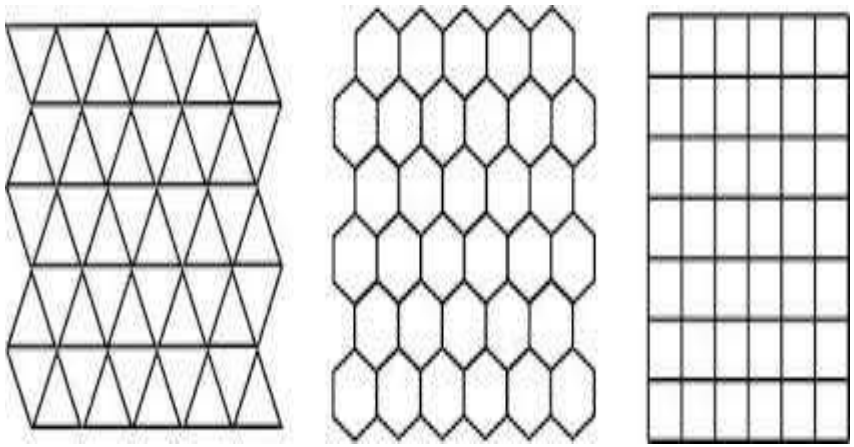
- Monohedral tiling
- Dihedral tiling
- Drawing tiling
- Reptiles

Monohedral tiling

- Based on single polygon
- **Types**

1. Regular tiling
2. Patterns
3. Cario tiling
4. Polymino
5. Polyiamond

Regular Tiling

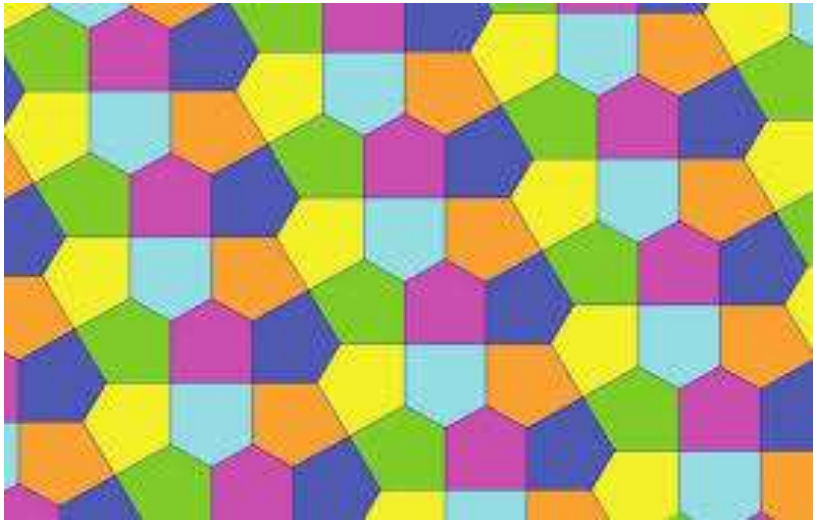


Patterns

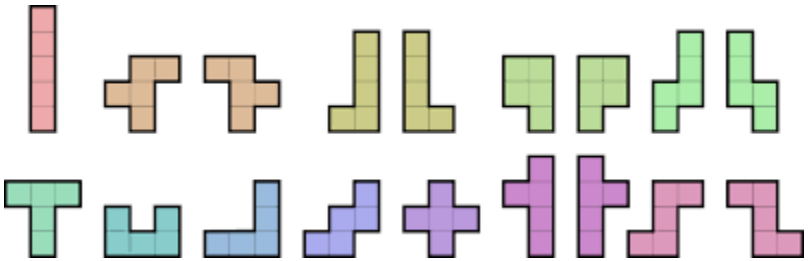
- Shifting the tessellation in particular direction

Cairo tiling

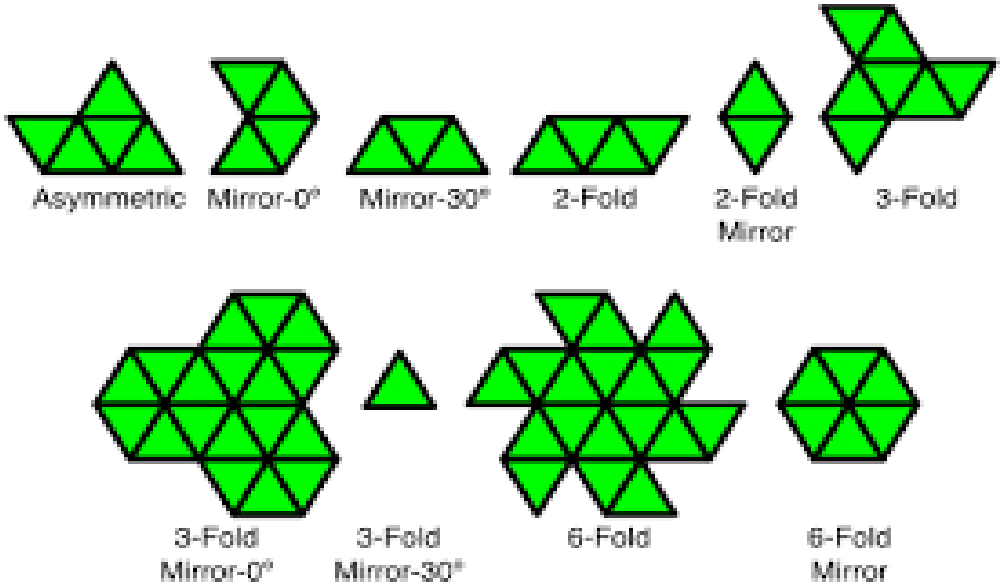
- Four pentagon fit together to form hexagon
- Used to tile the plane
- Many street in cairo,Egypt in this pattern



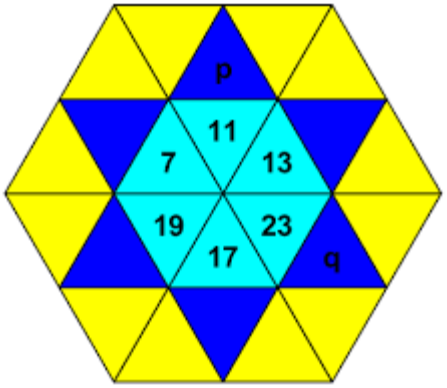
Polymino

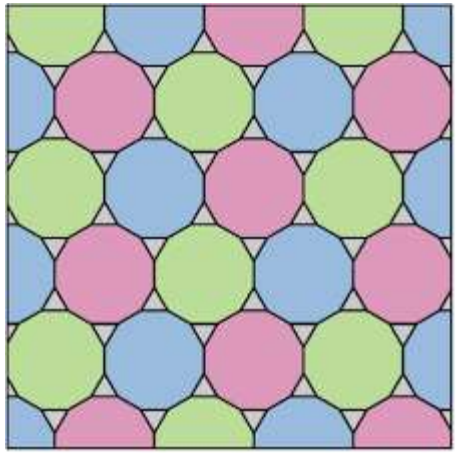


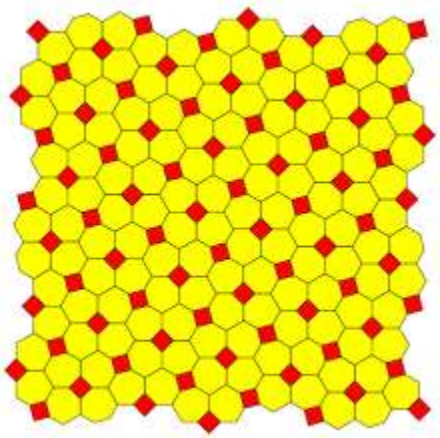
Polyaimonds



Dihedral Tiling

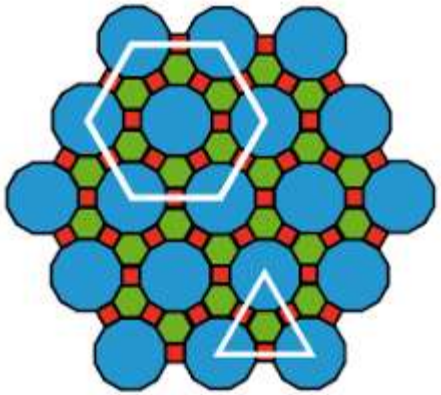






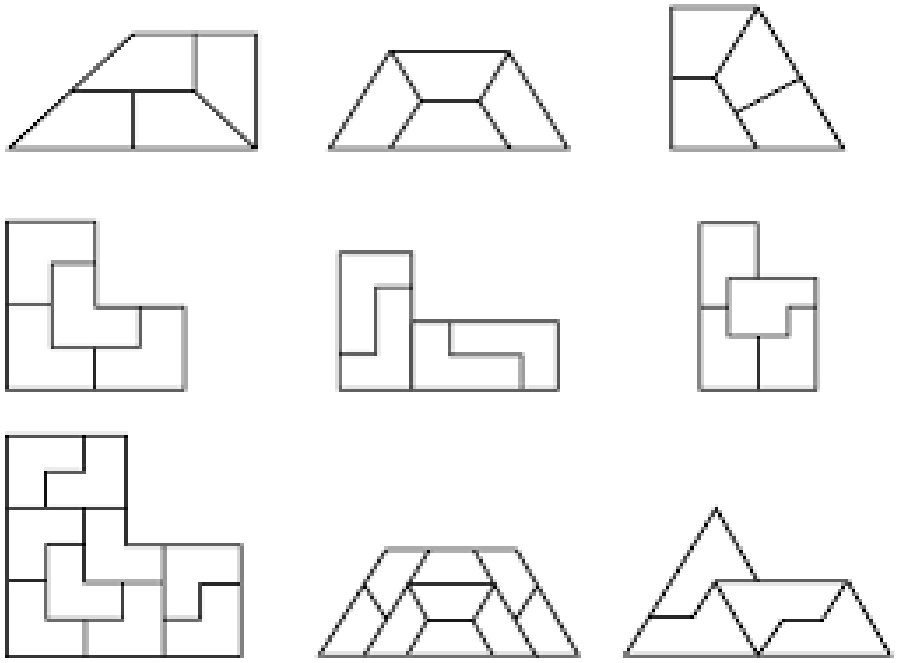
Drawing Tiling

- Large window setup
- Tiles grouped together into single figure
- Single figure drawn again and again
- Non periodic figure include
- Small to large and large to small



Reptiles

- Non periodic tiling
- Based on square, equilateral triangle



Application of tiles

Fractals

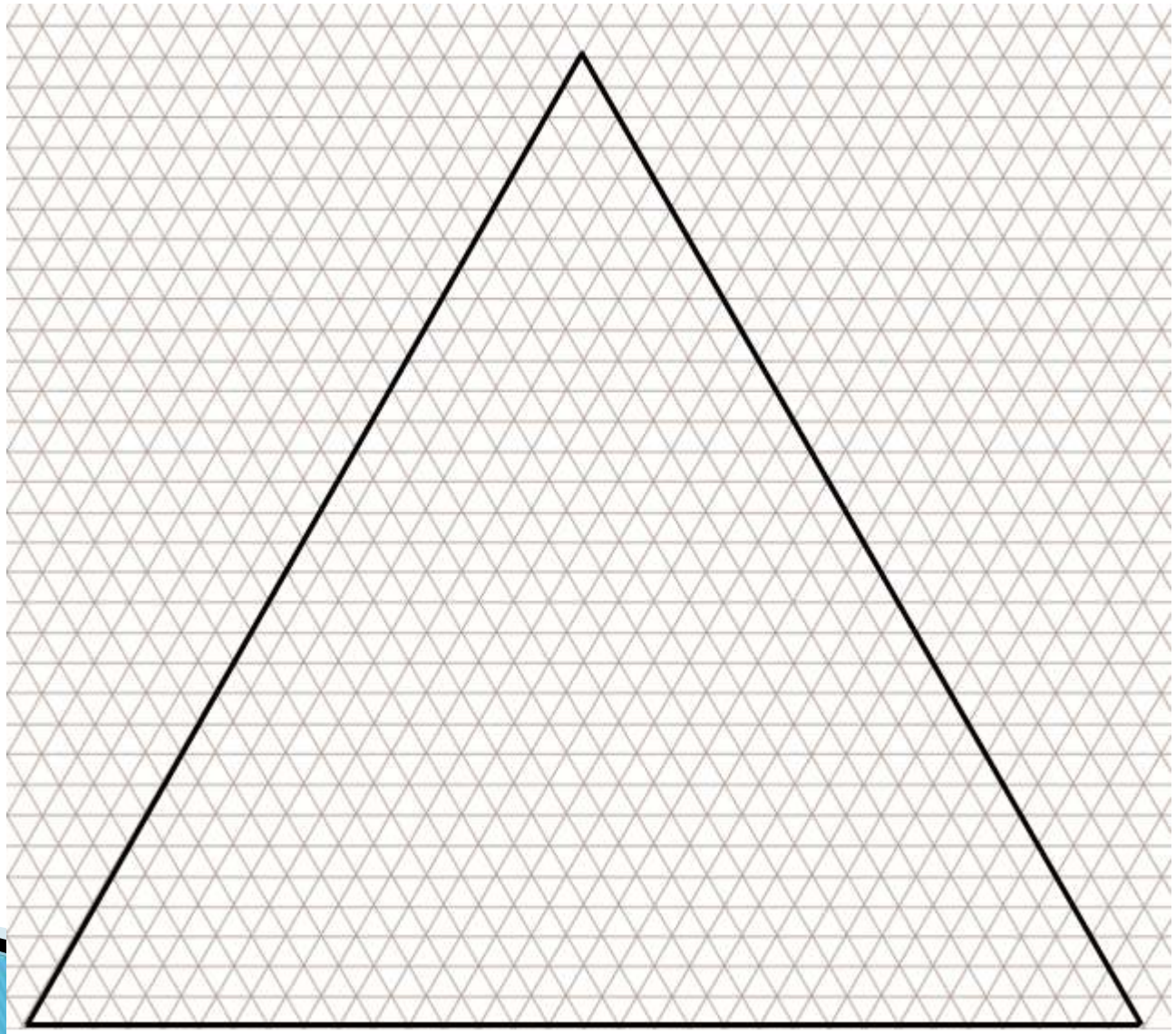
- A fractal is a never-ending pattern.
- Fractals are infinitely complex patterns that are self-similar across different scales.
- They are created by repeating a simple process over and over in an on-going feedback loop.

Types of Fractals

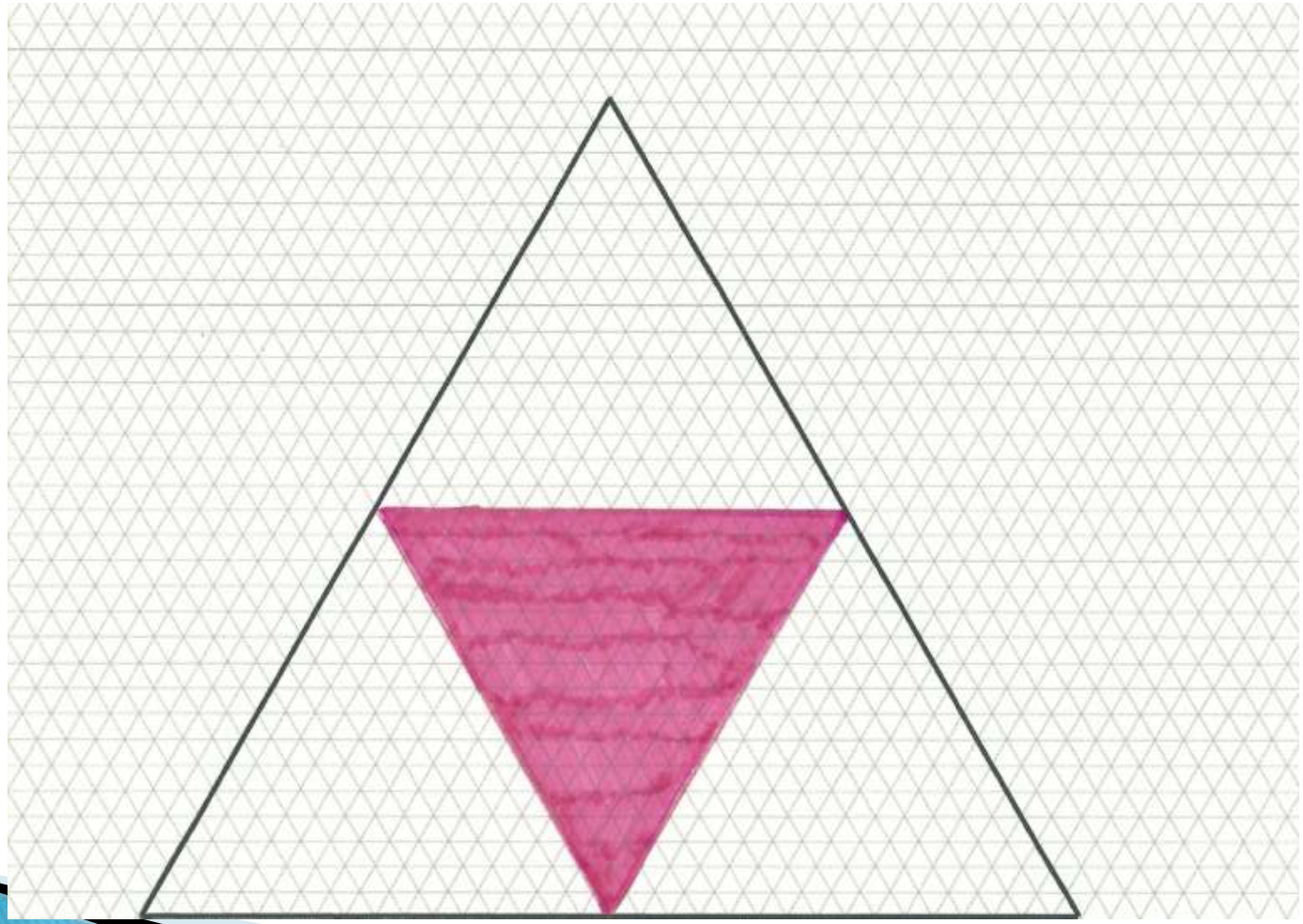
- Self Similar fractals
- Self Affine fractals
- Invariant fractals

Self Similar fractals

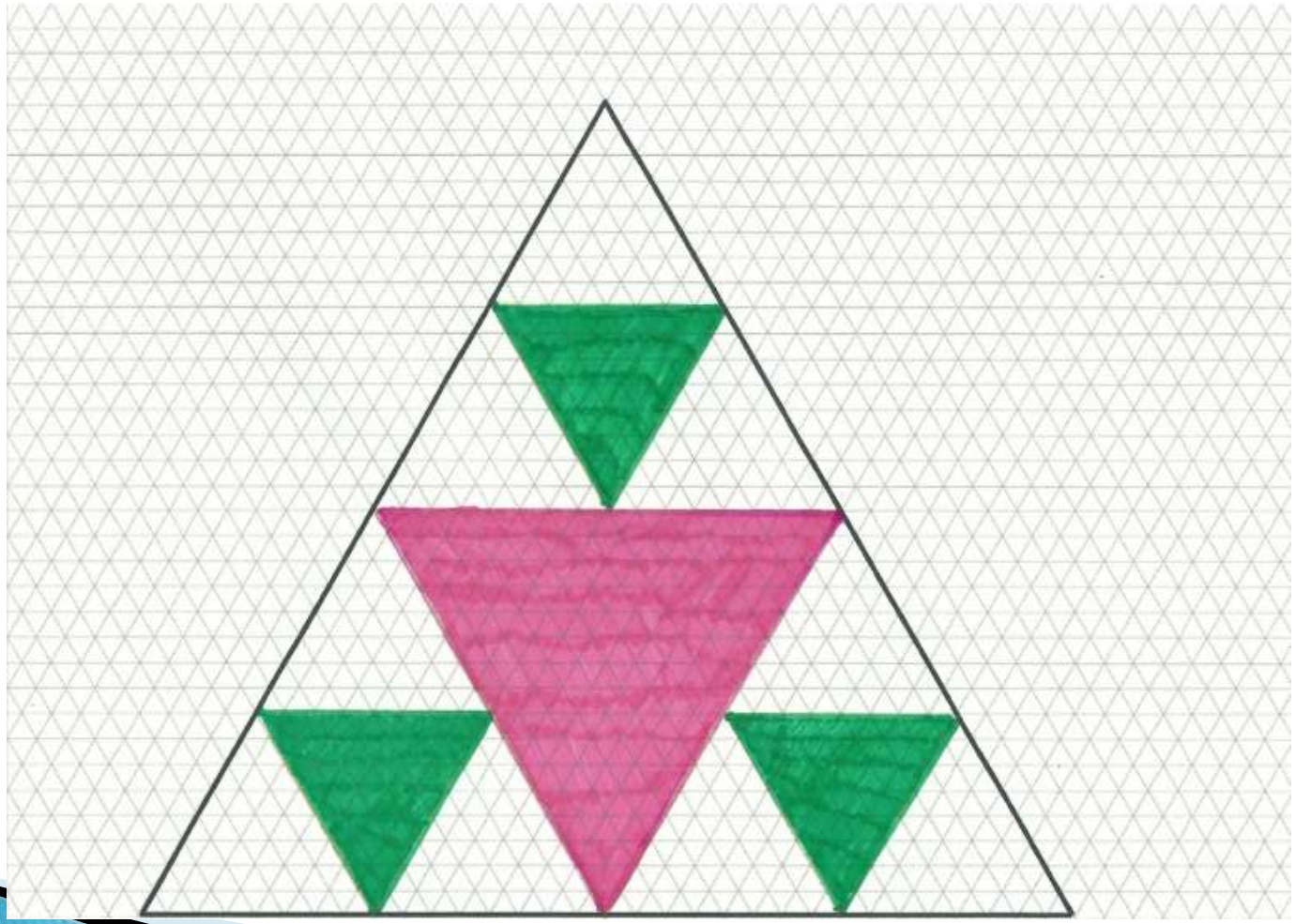
- Geometric figure is self similar
- Fractals appear identical at different scales



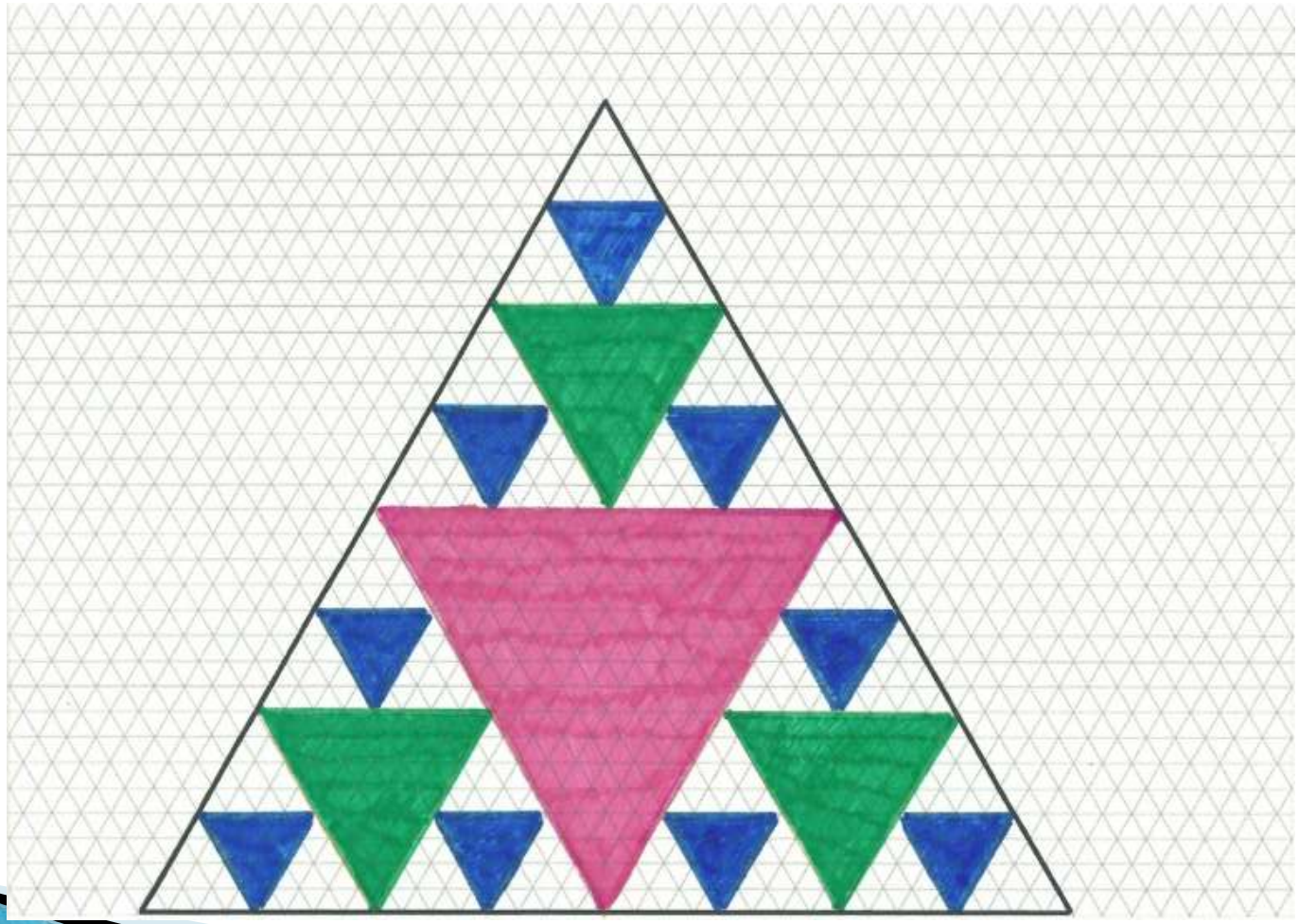
Iteration 1



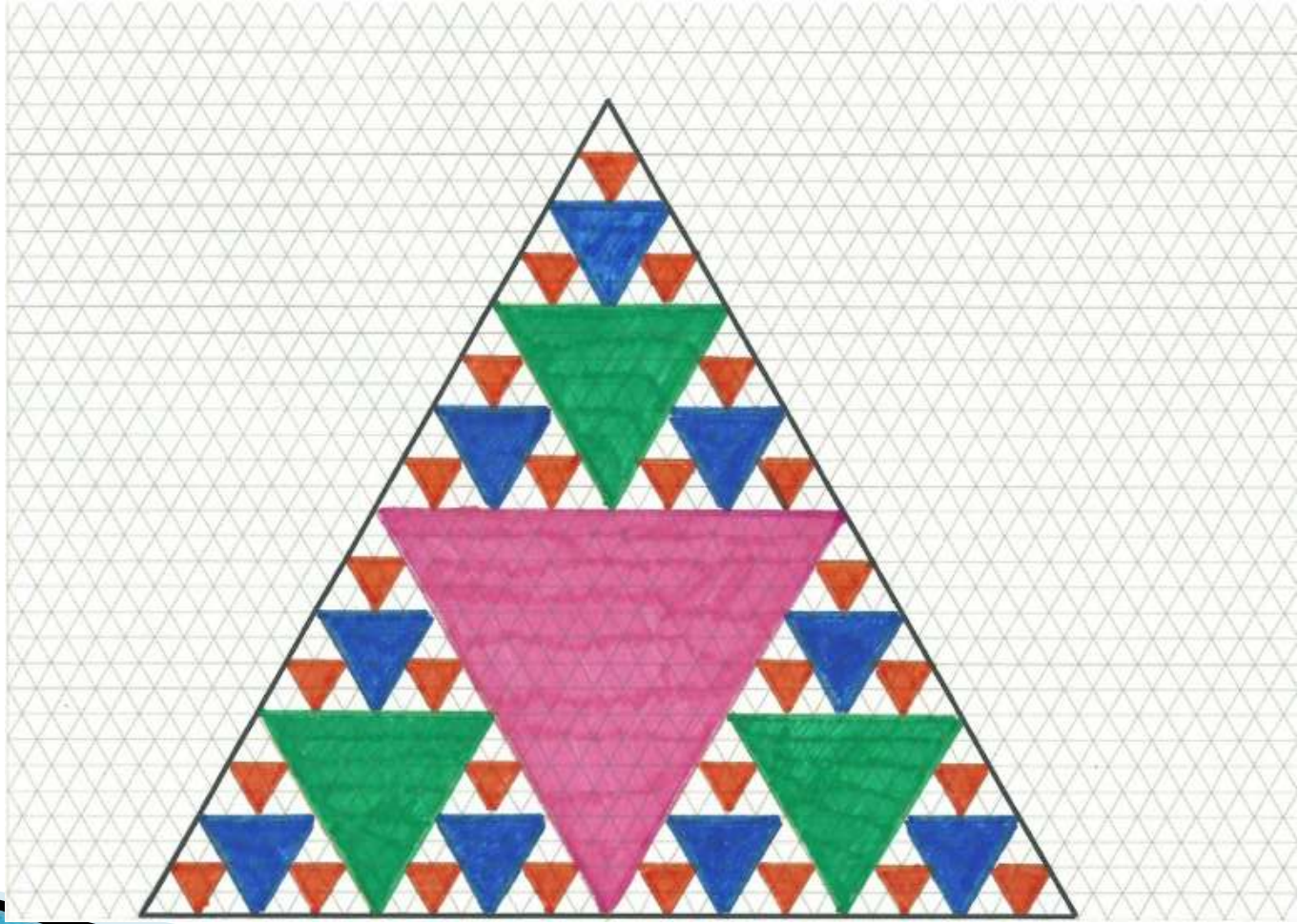
Iteration 2



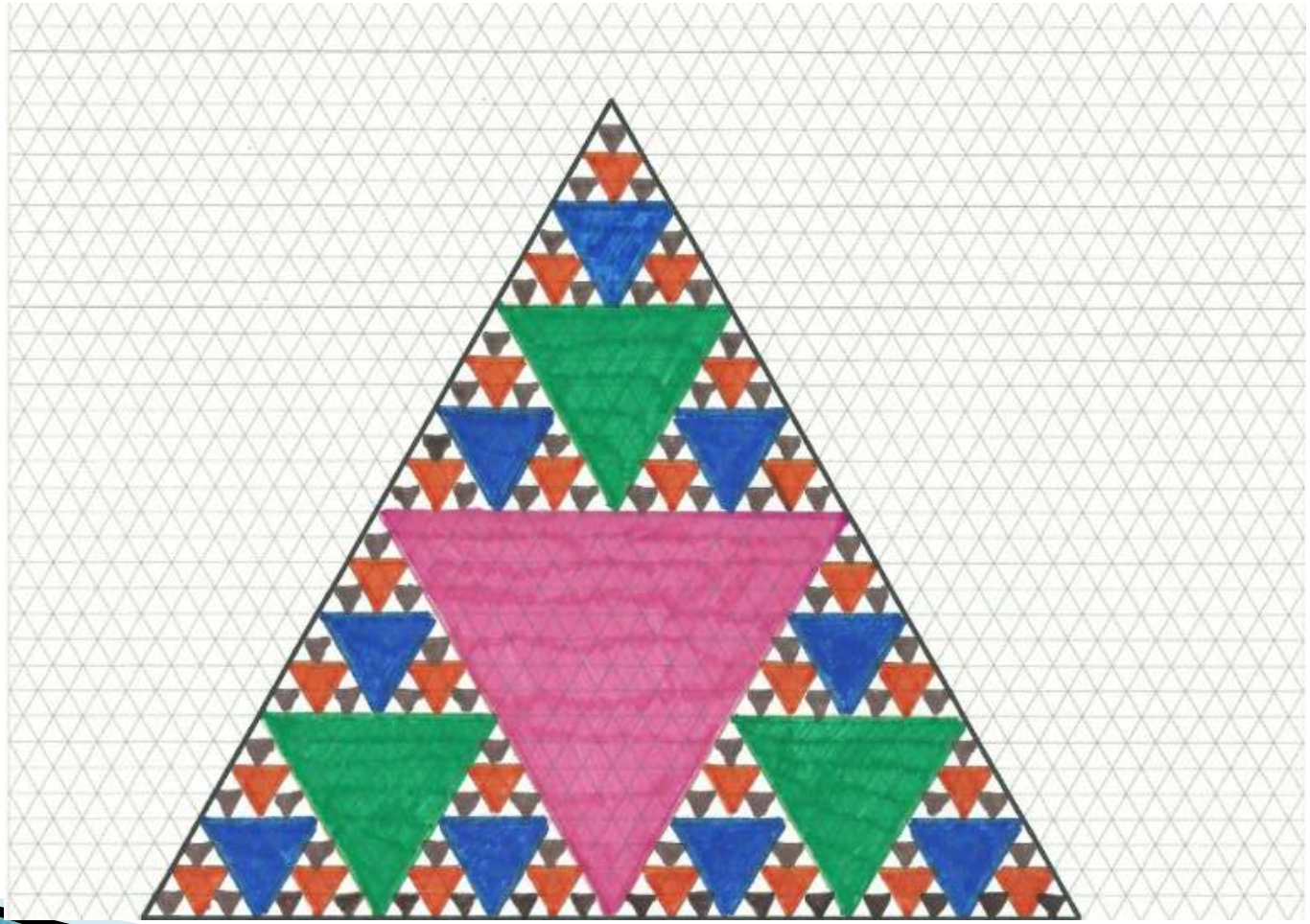
Iteration 3



Iteration 4



Iteration 5



Self similar fractal usage

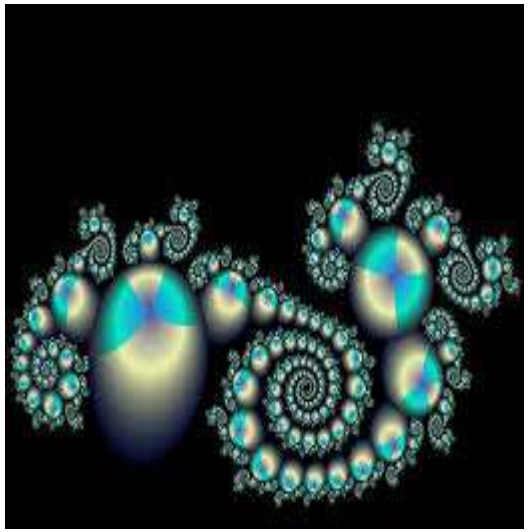
- Model trees,shrubs,plants

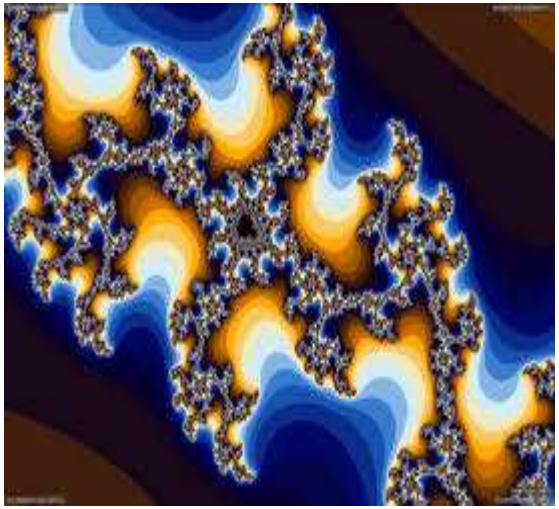
Self Affine fractals

- Fractal appear approximately identical at different scales
- Model water,clouds,terrain

Invariant fractals

- Non linear transformation





Recursively defined curves

- Curves created by iterations
- Formulas repeated with slightly different values over and over again

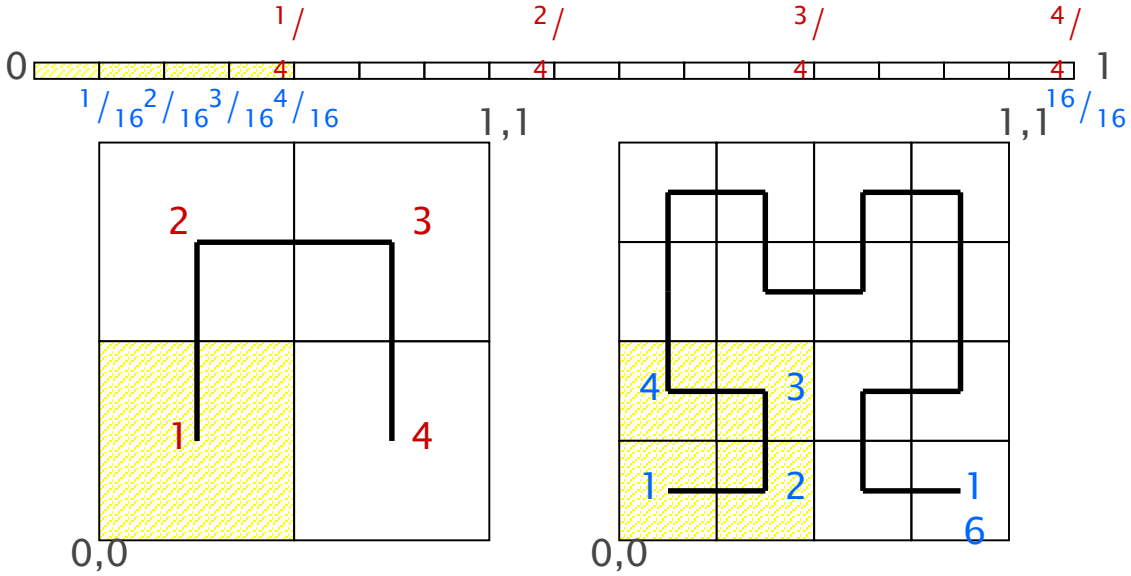
Types

- Hilberts Curve
- Koch Curve
- Dragon Curve
- Space filling Curve/Piano Curve
- C Curve

Hilberts Curve

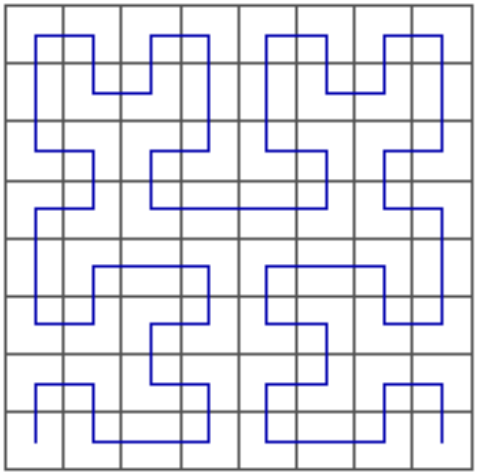
- It was described by the German mathematician David Hilbert in 1891.
- The Hilbert curve is a space filling curve.
- It visits every point in a square grid with a size of 2×2, 4×4, 8×8, 16×16, or any other power of 2.

The Hilbert curve: geometric generation



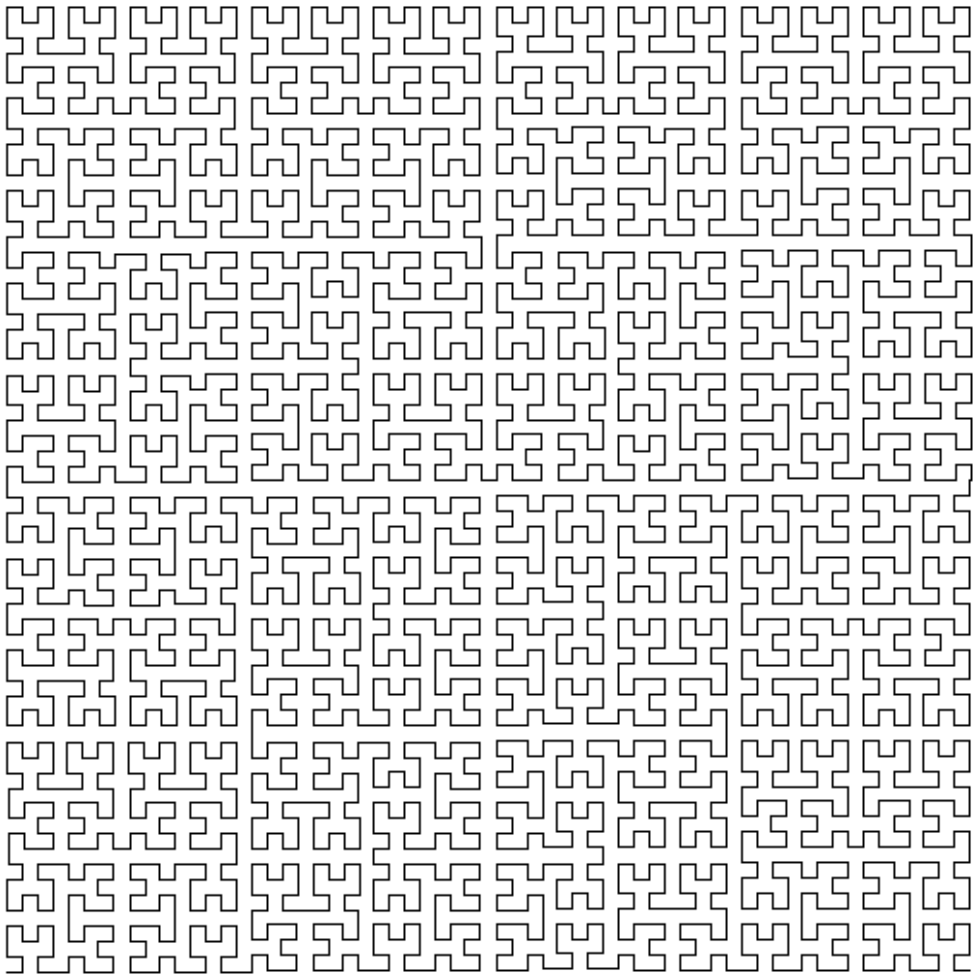
1st iteration

2nd iteration



3rd iteration

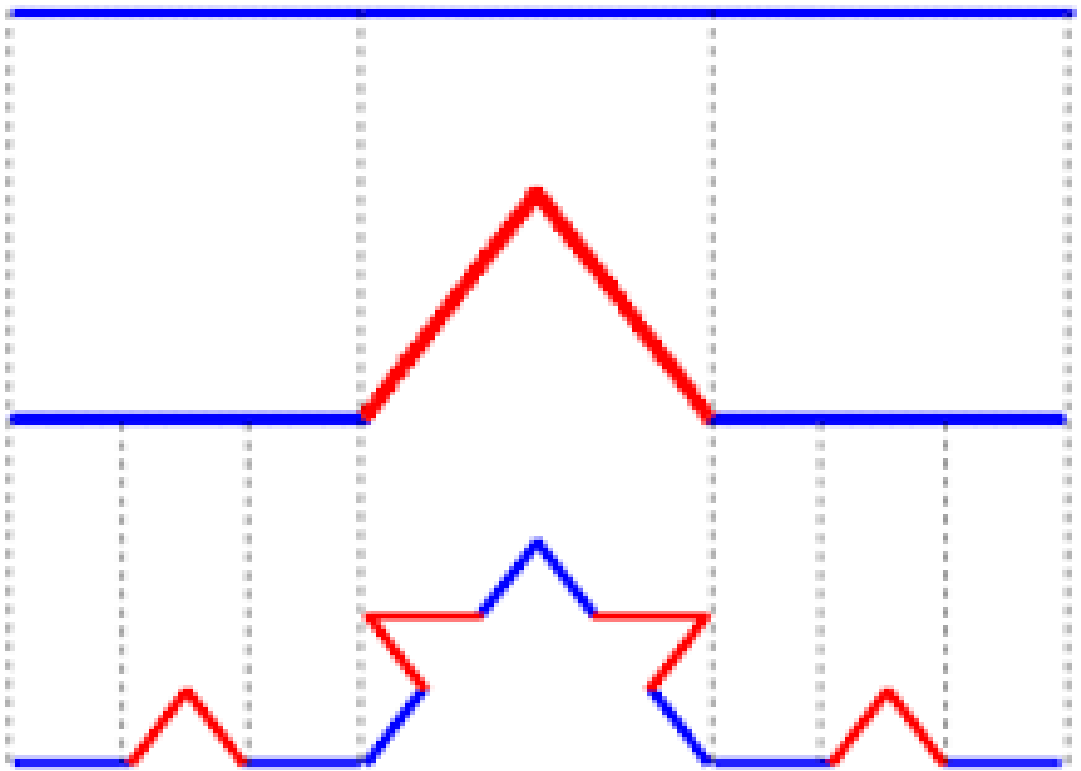
The Hilbert curve: geometric generation

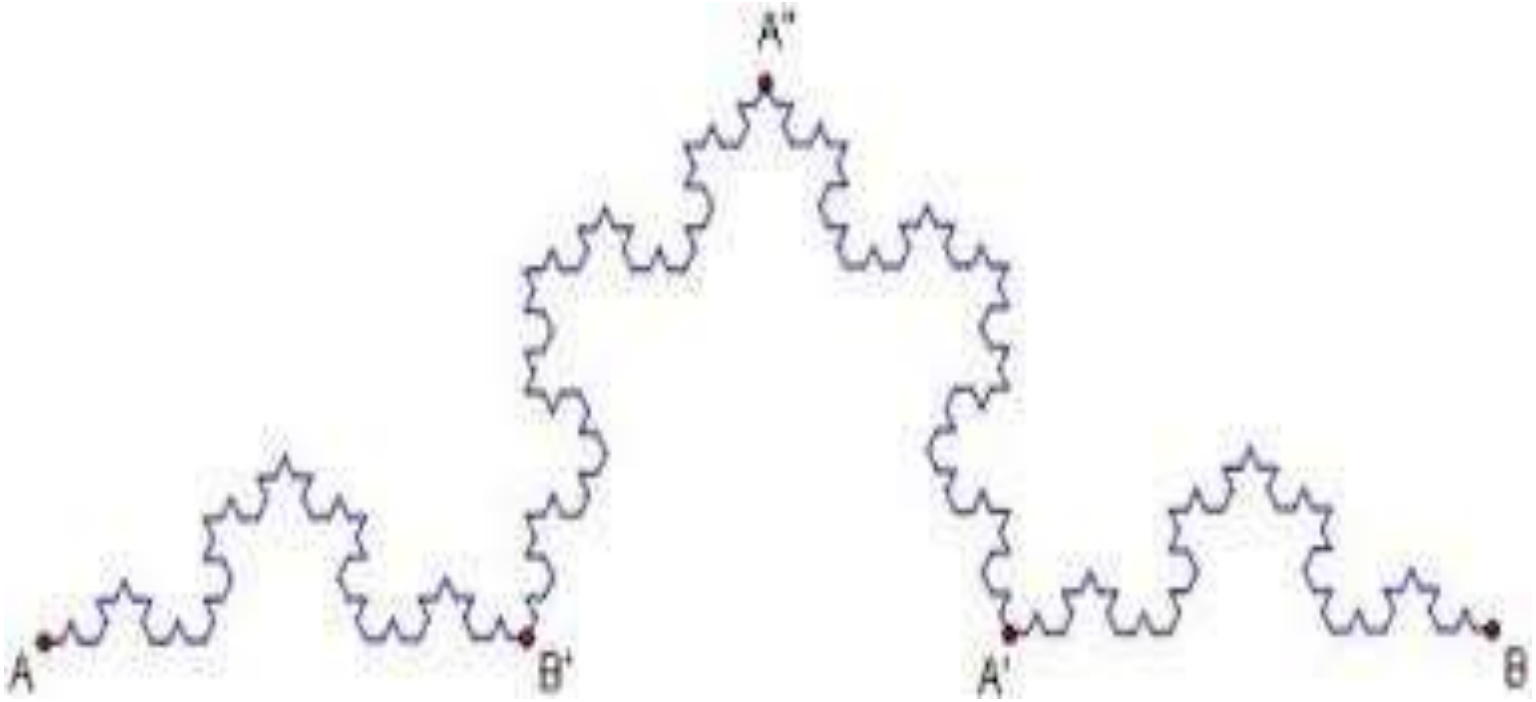


6th iteration

Koch Curve

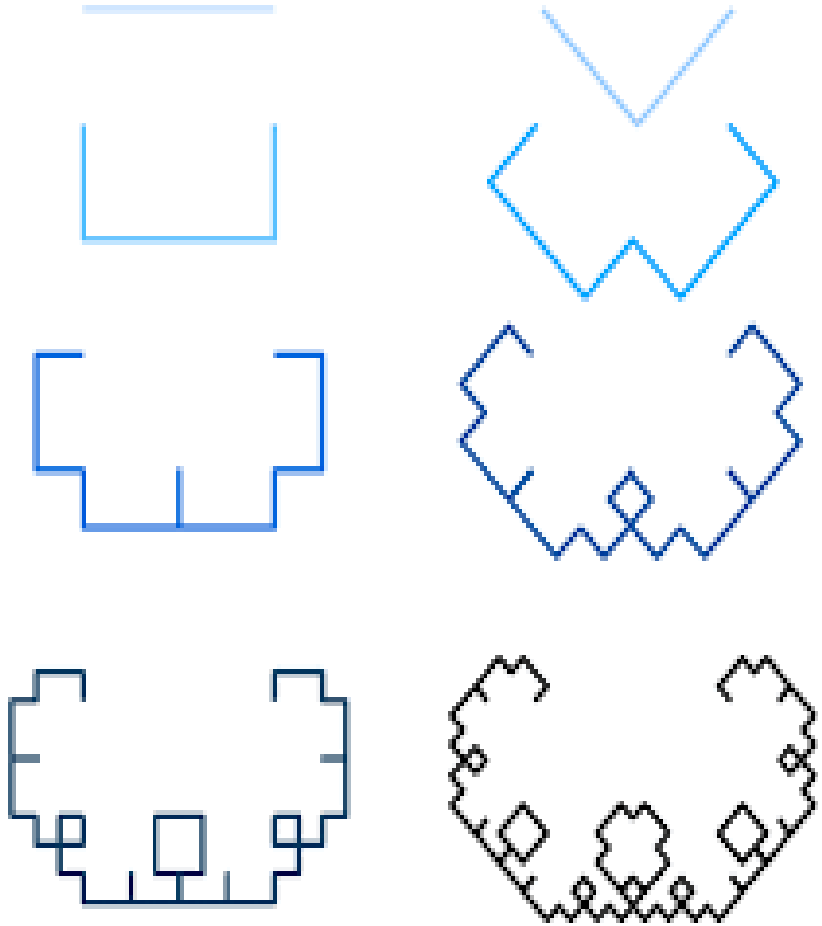
- Developed by Helga von Koch in 1904

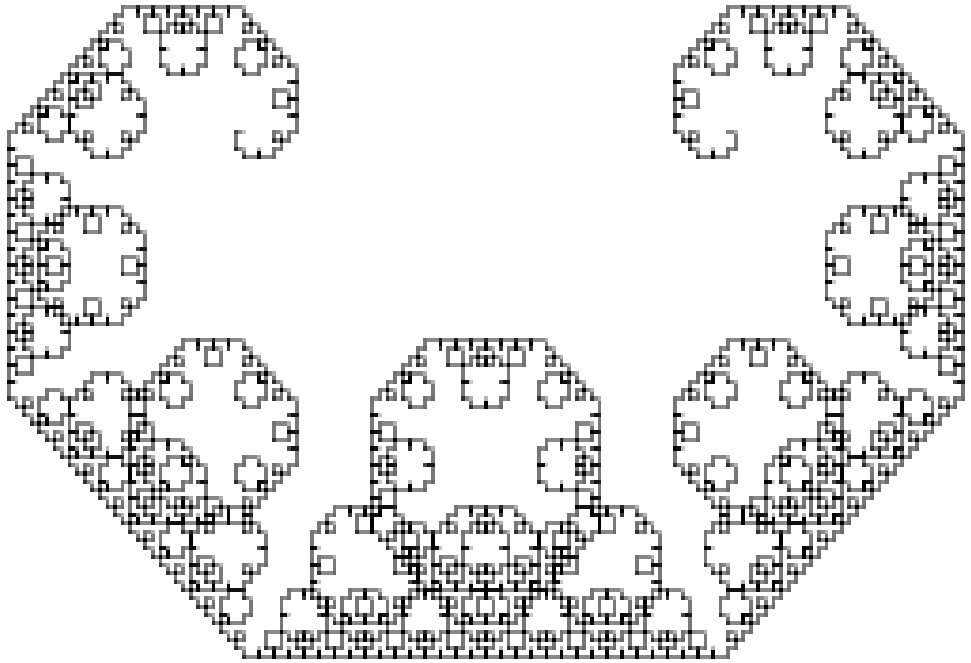



A''
A
B'
A'
B

C Curves

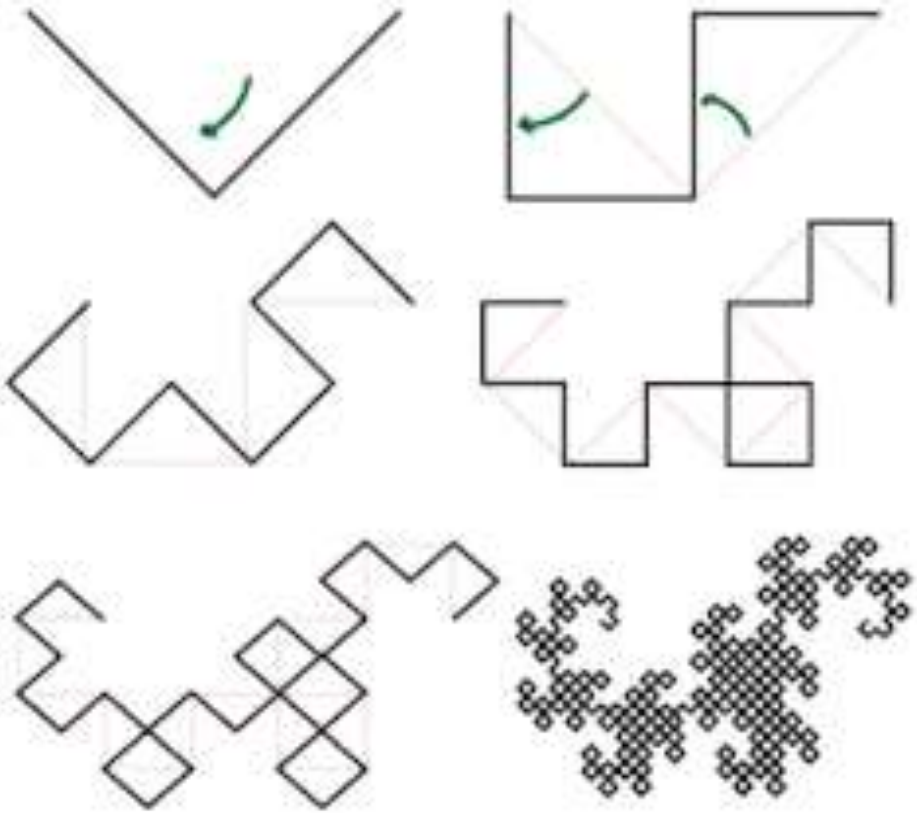
- Self similar fractals
- Described by Ernesto cesaro and Georg Faber in the year 1910

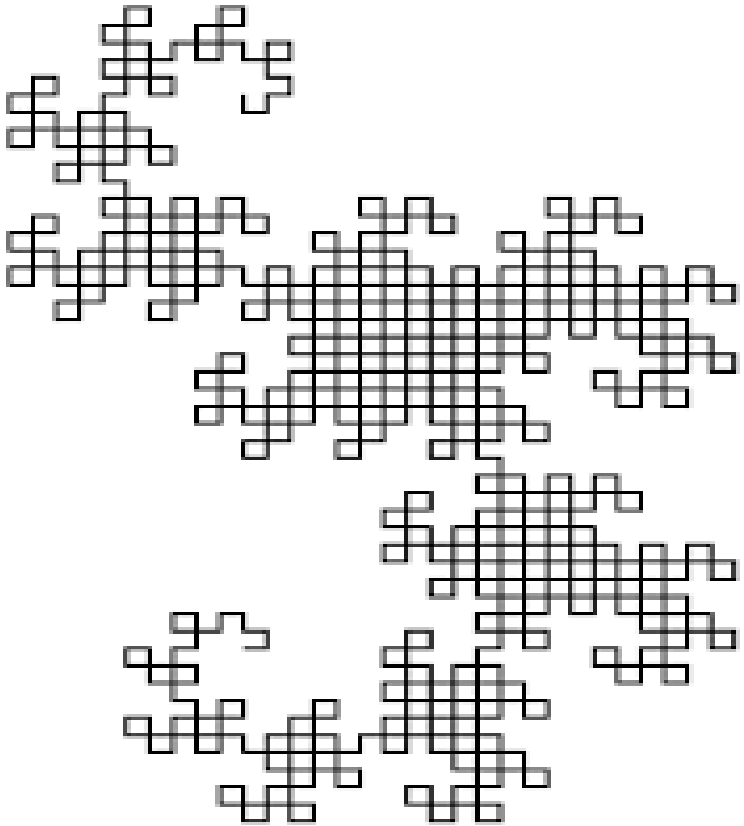




Dragon Curves

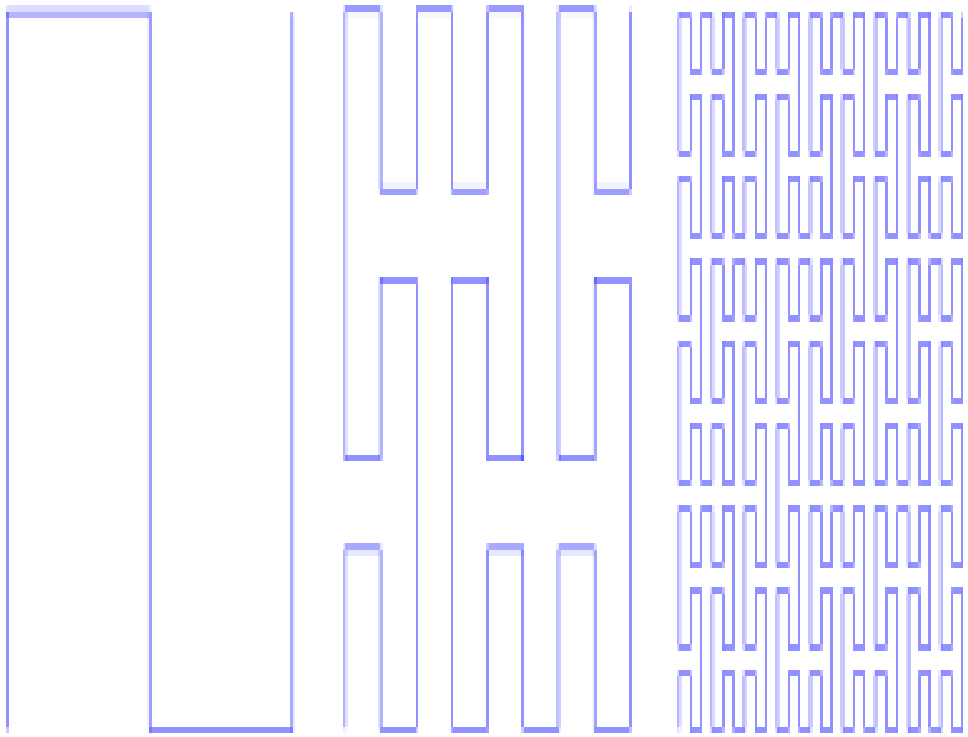
- Self similar fractal curves





Space filling curve/Peano curve

- Developed by Italian mathematician Guiseppe peano in 1890
- Space filling curve



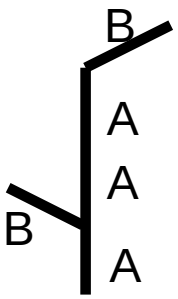
Grammar based models

- Structure defined by language
- Languages described by a collection of productions
- example, A->AA creates results of A, AA, AAAA,
- B->A[B] creates results of B, A[B], AA[B], etc.

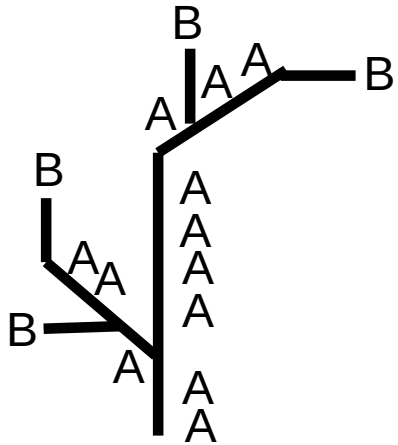
Advanced Modeling

Graftals

- Grammar based models…
  - [ ] for left branches
  - ( ) for right branches
  - A -> AA and B -> A[B]AA(B)
  - create a 2nd generation of:
    AA[A[B]AA(B)]AAAA(A[B]AA(B))



First Generation



Second Generation

Advanced Modeling

Graftals

- Grammar based models…
  - …use biological productions to simulate plants in development
  - …describe the topology of plants
  - …also describe the shape including the directions of branches and the arrangement of leaves

Advanced Modeling

- To simulate the growth of plants using languages include information on...
  - ...the current age
  - ...the growth rate of each segment
  - ...the probabilities of death, dormancy, growth
  - ...the shape (depending on type and age)
  - ...the branch angles (depending on type and age)
  - ...the color and texture of each segment

Advanced Modeling

Graftals

- Pseudo code simulates the growth of plants using graftals:
  - For (each moment in time)
    - For (each bud that is still alive)
    - Determine whether the bud dies, is dormant, or grow
    - If (the bud does not die)
    - If (the bud is not dormant)
    - Create a portion of a stem, determining its direction, position, color, texture;

    Create a new bud;

Advanced Modeling

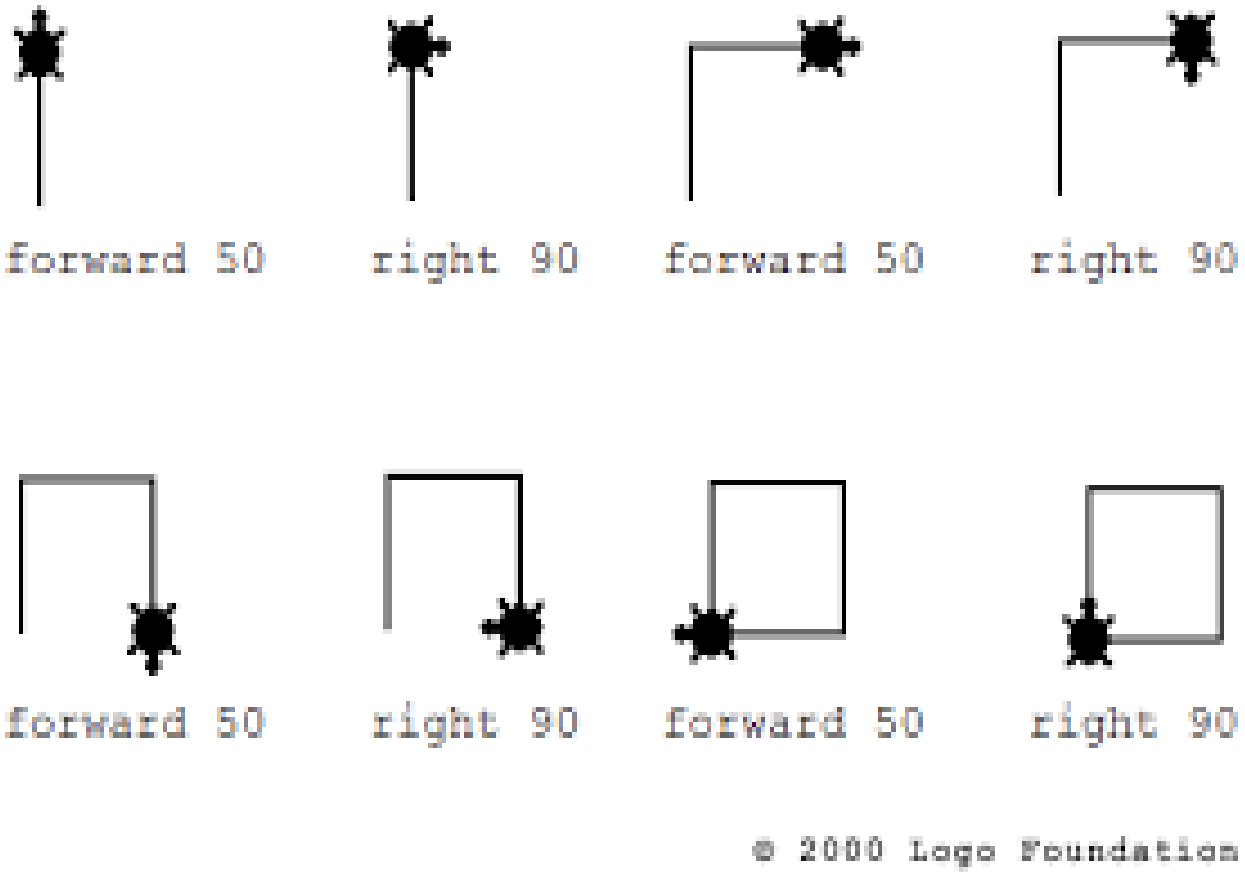
Particle Systems

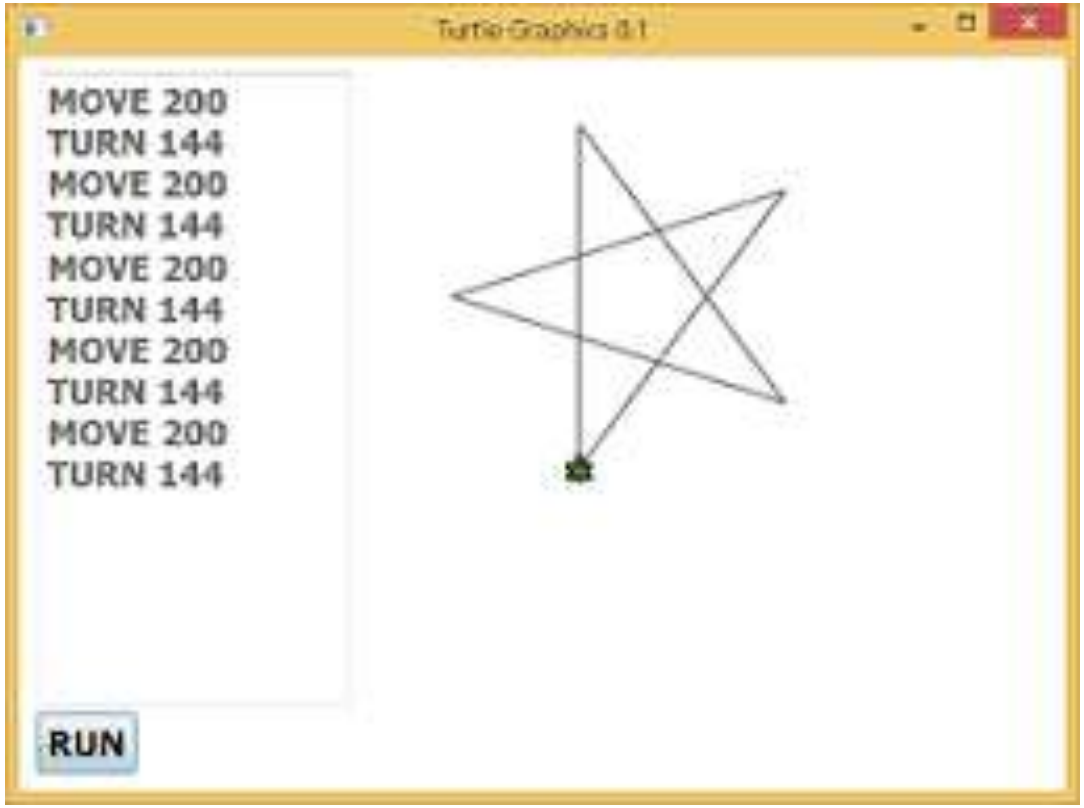
- Particle systems…
  - …can be used to simulate fire, clouds, water, fog, smoke, fireworks, trees, and grass
  - …are particularly useful for animating objects instead of just simulating static objects

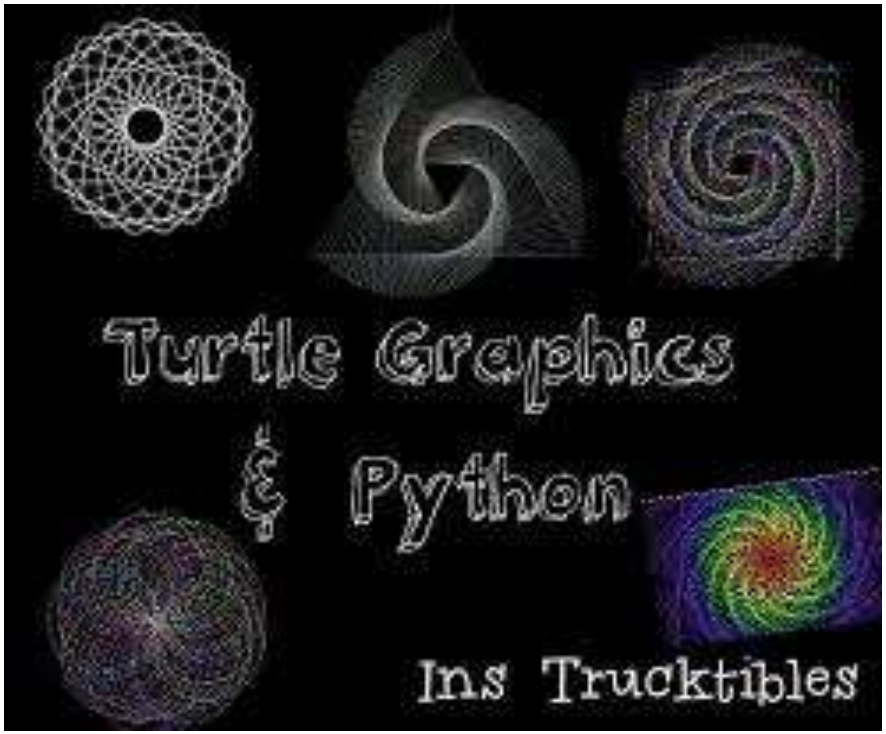

Hindi
Spanish
English
INDO-
EUROPEAN
OLD WORLD
LANGUAGE FAMILIES

Turtle Graphics

- Logo programming language
- Developed by feurzig & seymour papert in 1966
- Popular graphics language for kids




Turtle Graphics 0.1
MOVE 200
TURN 144
MOVE 200
TURN 144
MOVE 200
TURN 144
MOVE 200
TURN 144
MOVE 200
TURN 144
RUN


Turtle Graphics
& Python
Ins Trucktibles


BARRIER ISLAND
CENTER