L-SYSTEMS

L-systems

Lindenmeyer systems, developed in 1968, to describe the growth process of living organisms such as branching patterns of plants.

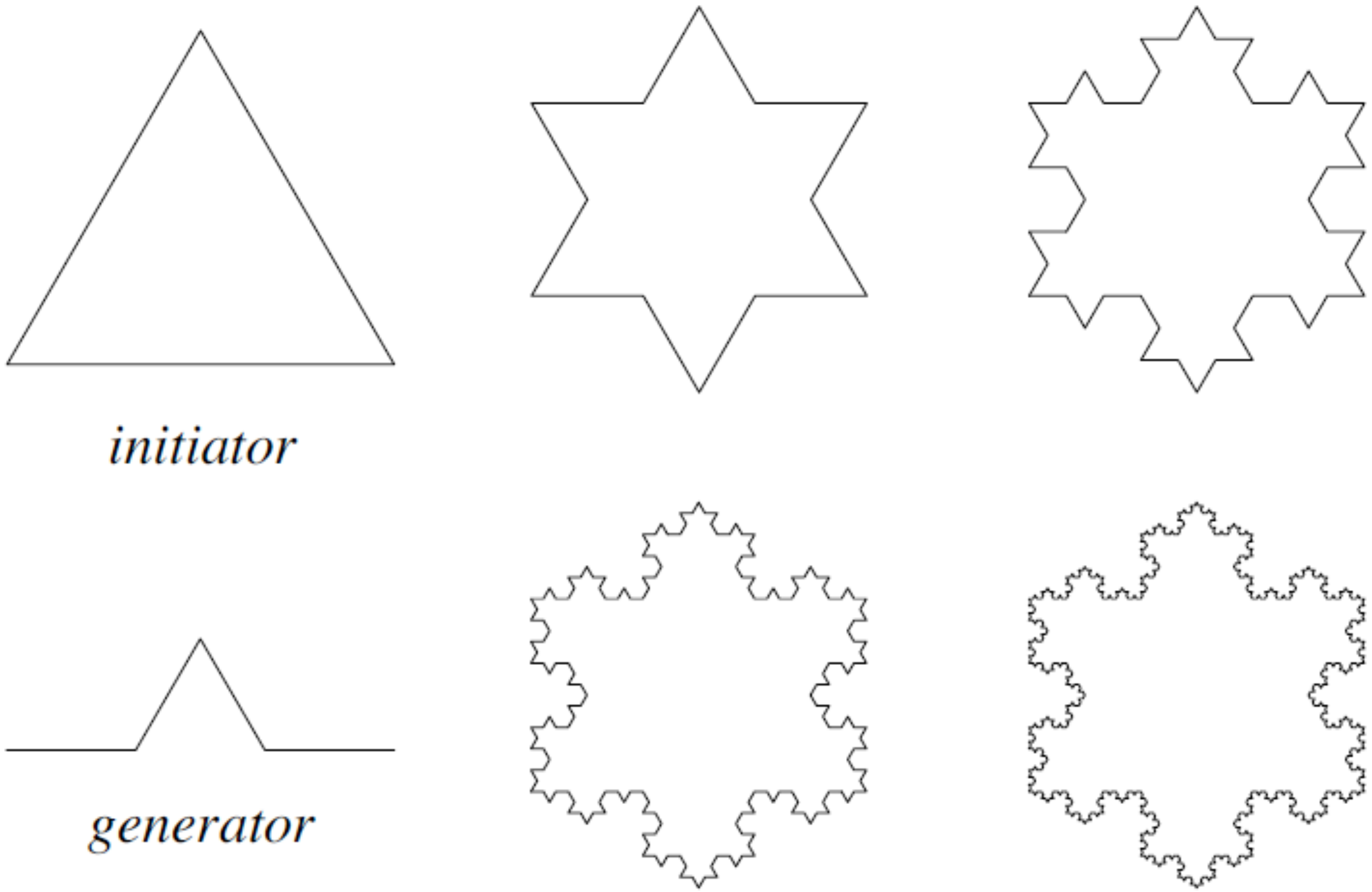
Algorithmic Beauty of Plants

http://algorithmicbotany.org/papers/abop/abop.pdf_

http://algorithmicbotany.org/papers/

- Axiom
- Vocabulary
- Production applied in parallel

Snowflakes - Koch curve



L-systems as turtles

F draw forward

G move forward

+ turn right through fixed angle

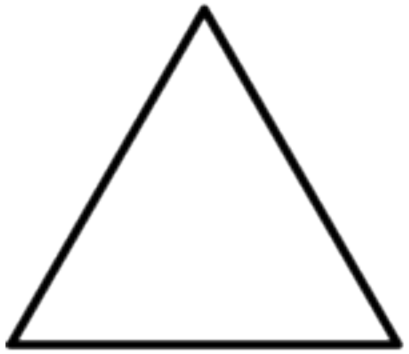
– turn left through fixed angle

[save turtles current position (start a new branch)

] remove last stored state and use to restore

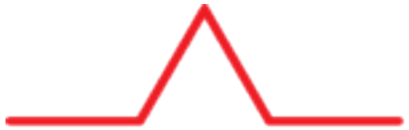
| move and draw turtle by a length

Koch snowflake as an L-system



Initial shape

F++F++F



Generator

F-F++F-F

String Replacement

```
string S = "F++F++F";
string R = "F-F++F-F";
while(N > 0)
   {
     Print("N = {0}", N);
     N--;
     S = S.Replace("F", R);
   }
```

Geometry Output

F : draw forward

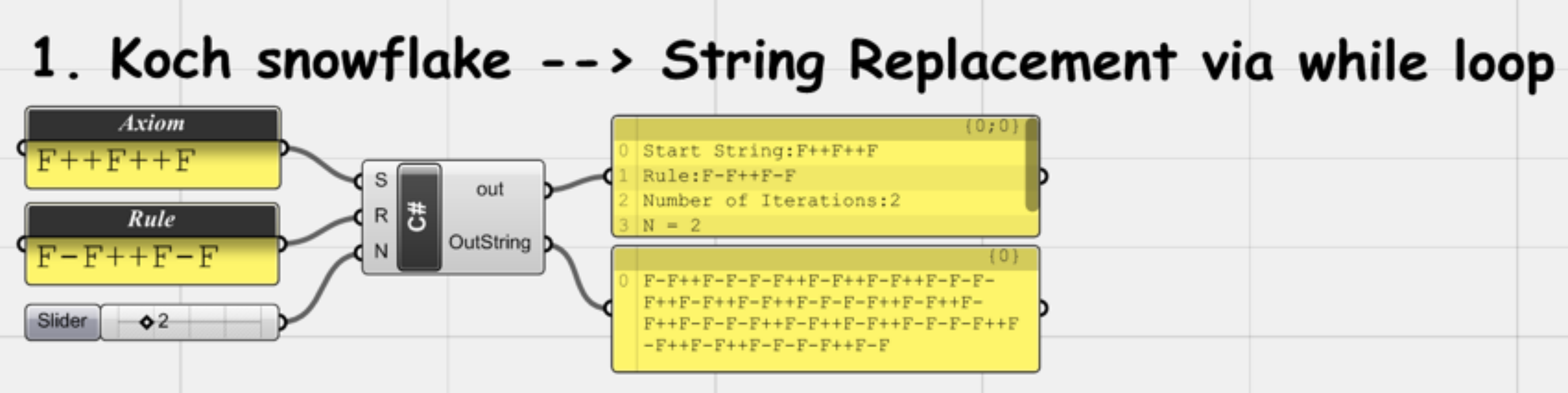
+ : turn + 60°

− : turn - 60°

Koch snowflake _formatted string output

String Replacement

```
string S = "F++F++F";
string R = "F-F++F-F";
while(N > 0)
  {
    Print("N = {0}", N);
    N--;
    S = S.Replace("F", R);
  }
```



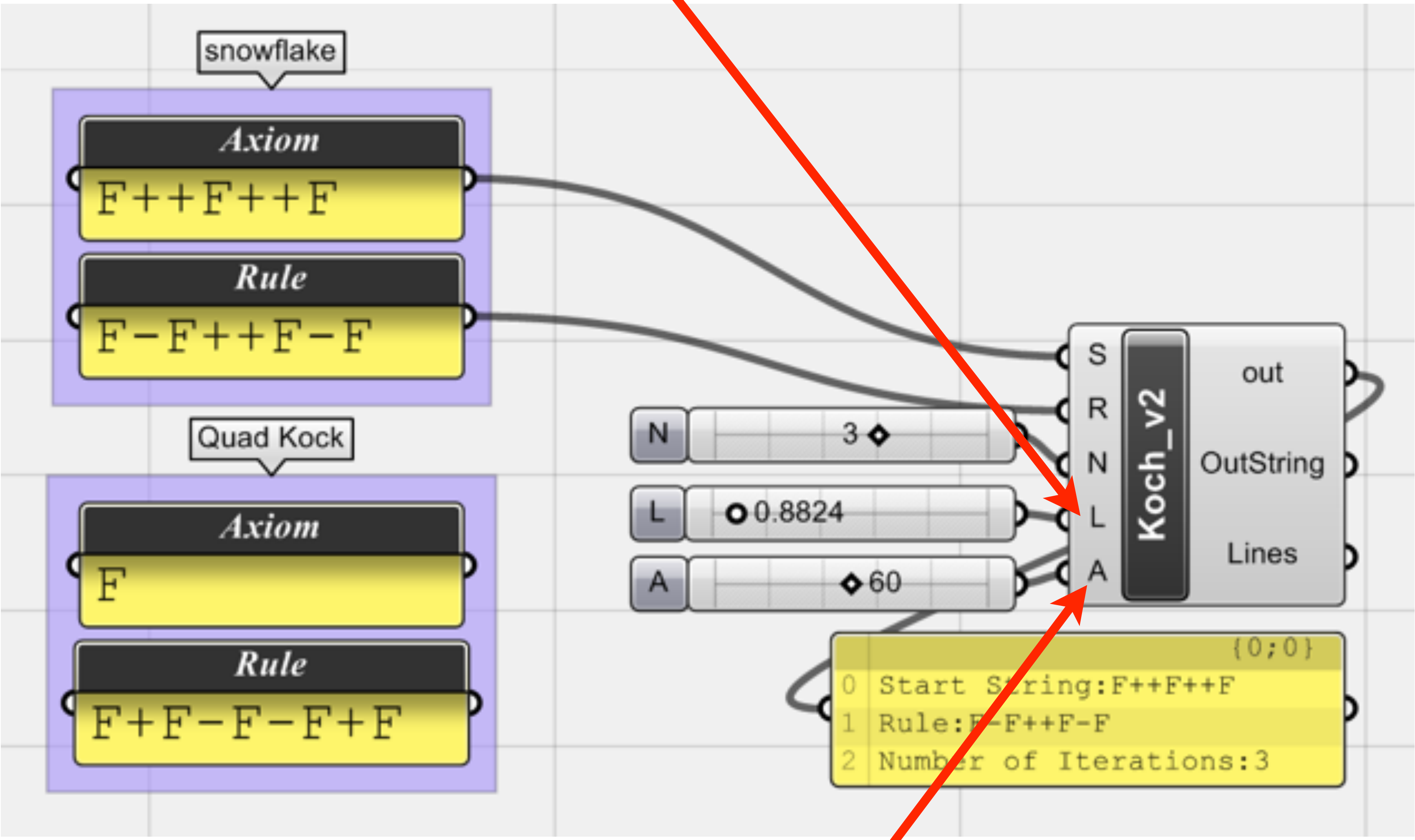
1. Koch snowflake --> String Replacement via while loop

```
0 Start String:F++F++F
1 Rule:F-F++F-F
2 Number of Iterations:2
3 N = 2
```

```
0 F-F++F-F-F-F++F-F++F-F++F-F-F-
  F++F-F++F-F++F-F-F-F++F-F++F-
  F++F-F-F-F++F-F++F-F++F-F-F-F++F
  -F++F-F++F-F-F-F++F-F
```

Koch snowflake: modularization

Unit Length of Drawing



snowflake

Axiom: F++F++F

Rule: F-F++F-F

Quad Kock

Axiom: F

Rule: F+F-F-F+F

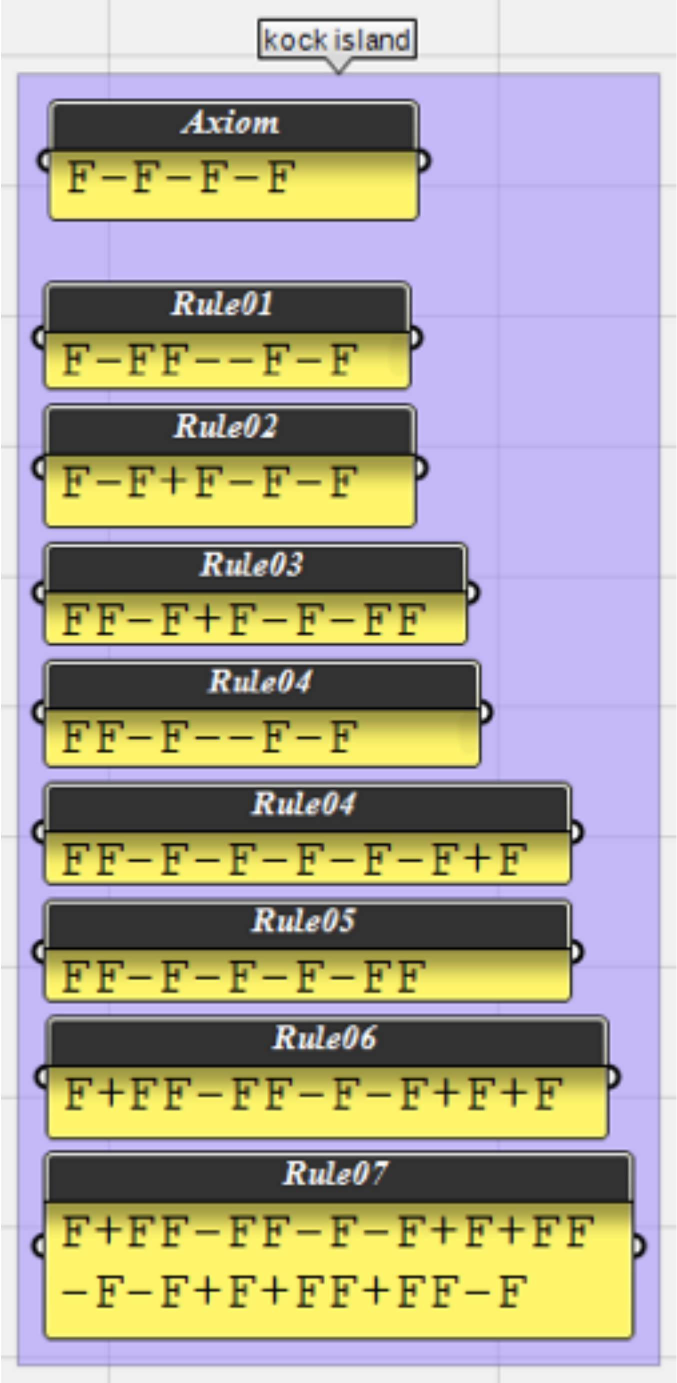
N: 3

L: 0.8824

A: 60

```
{0;0}
0 Start String:F++F++F
1 Rule:F-F++F-F
2 Number of Iterations:3
```

Rotation Angle



kock island

Axiom: F-F-F-F

Rule01: F-FF--F-F

Rule02: F-F+F-F-F

Rule03: FF-F+F-F-FF

Rule04: FF-F--F-F

Rule04: FF-F-F-F-F-F+F

Rule05: FF-F-F-F-FF

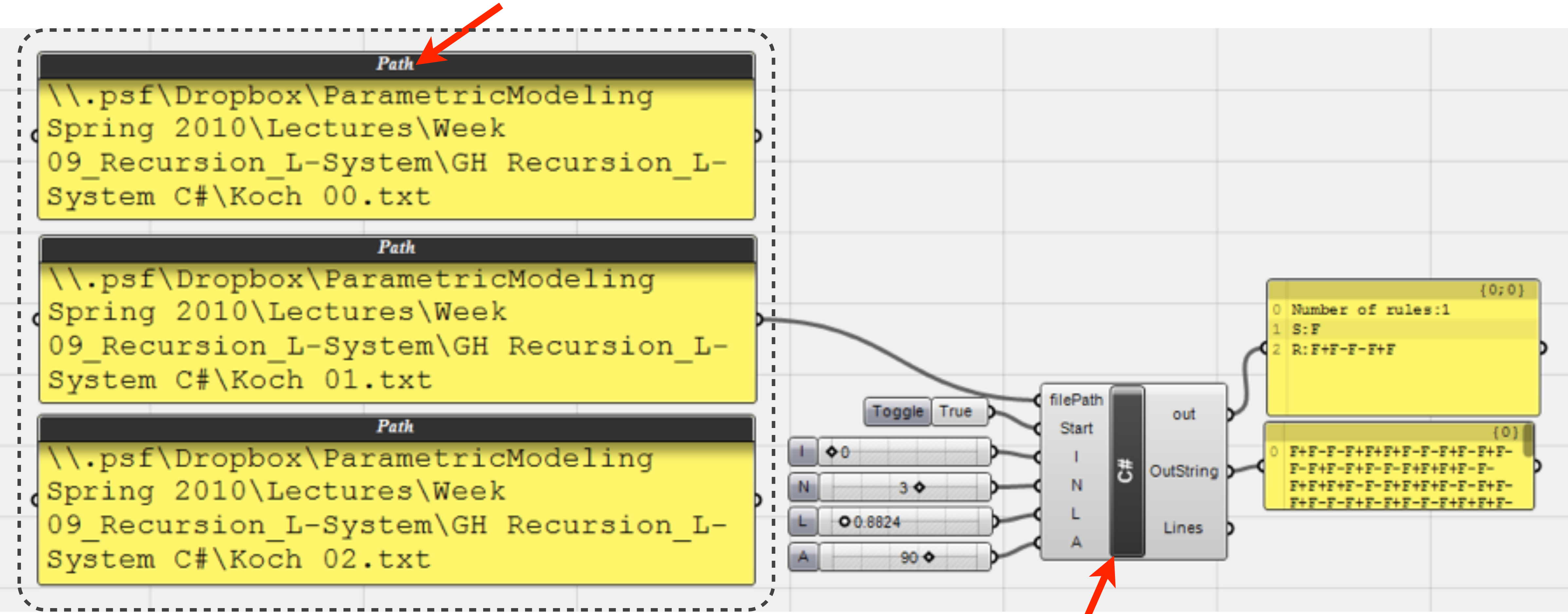
Rule06: F+FF-FF-F-F+F+F

Rule07: F+FF-FF-F-F+F+FF
-F-F+F+FF+FF-F

More rules/variations

Koch snowflake _StreamReader

Path of the file



Path

\\.psf\Dropbox\ParametricModeling Spring 2010\Lectures\Week 09_Recursion_L-System\GH Recursion_L-System C#\Koch 00.txt

Path

\\.psf\Dropbox\ParametricModeling Spring 2010\Lectures\Week 09_Recursion_L-System\GH Recursion_L-System C#\Koch 01.txt

Path

\\.psf\Dropbox\ParametricModeling Spring 2010\Lectures\Week 09_Recursion_L-System\GH Recursion_L-System C#\Koch 02.txt

```
{0;0}
0 Number of rules:1
1 S:F
2 R:F+F-F-F+F
```

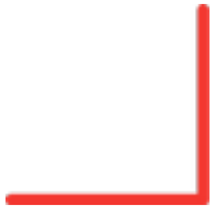
```
{0}
0 F+F-F-F+F+F+F-F-F+F-F+F-
  F-F+F-F+F-F-F+F+F+F-F-
  F+F+F+F-F-F+F+F+F-F-F+F-
  F+F-F-F+F-F+F-F-F+F+F+F-
```

C# component reads the files and generates output

L-System : Dragon curve

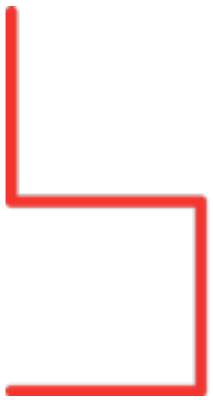
variables : X Y F
constants : + –
start : FX
ruleX : X+YF+
ruleY : –FX–Y
angle : 90°

F : draw forward
+ : turn + 90°
– : turn - 90°



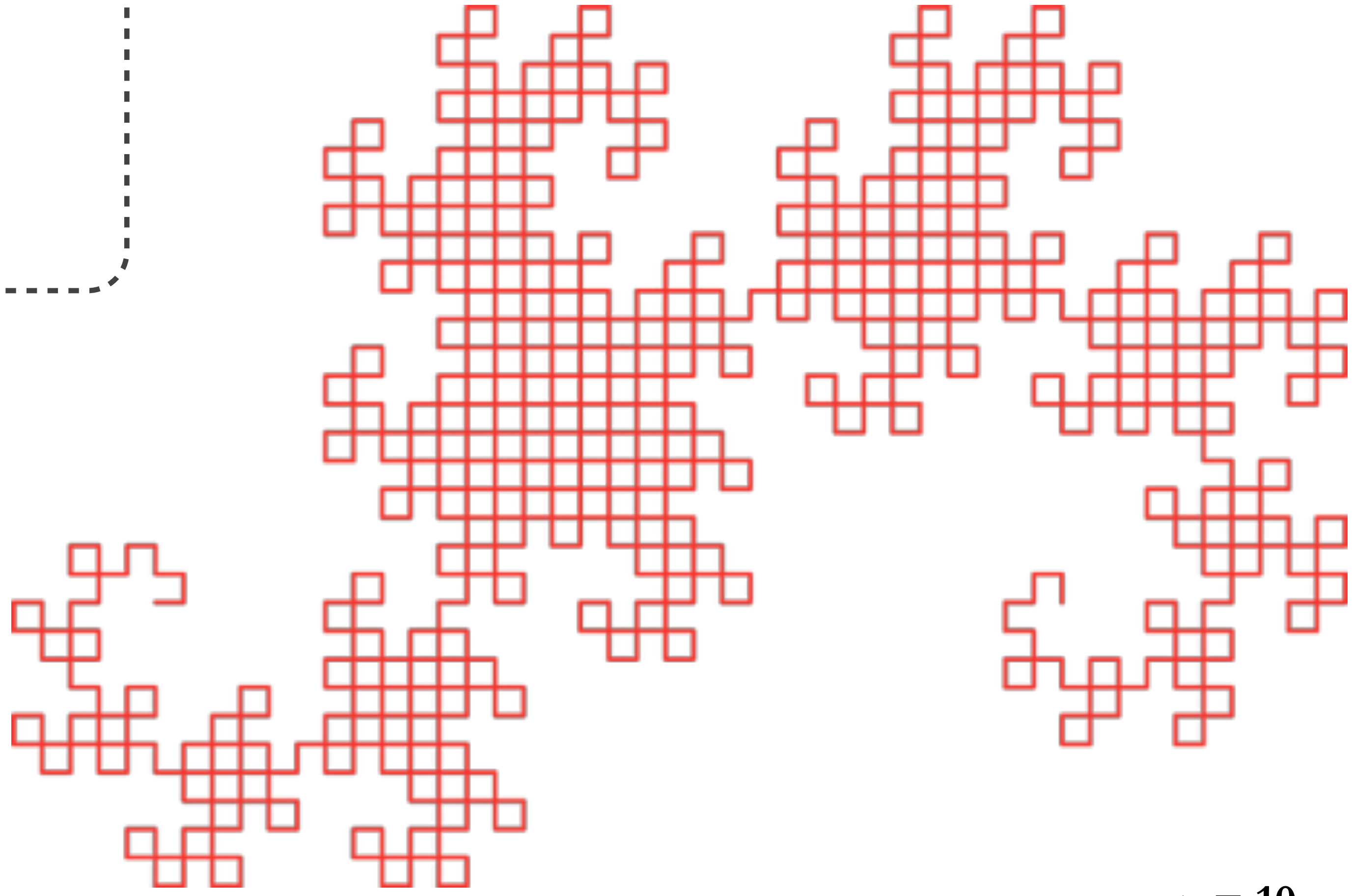
n = 1

FX+YF+



n = 2

FX+YF++-FX-YF+



n = 10

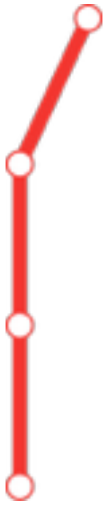
```
FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF+
+-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF+
+-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF
+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+
+-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF+
+-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF
+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF
+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+
+-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF+
+-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF+
+-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF
+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF
+--FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF
++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF
+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF
+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+
+-FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF++-FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF+
+-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF++-FX+YF++-FX-YF+--FX+YF+--FX-YF+--FX+YF++-FX-YF++-FX+YF+--FX-YF+--FX+YF++-FX-YF+--FX+YF+--FX-YF+
```

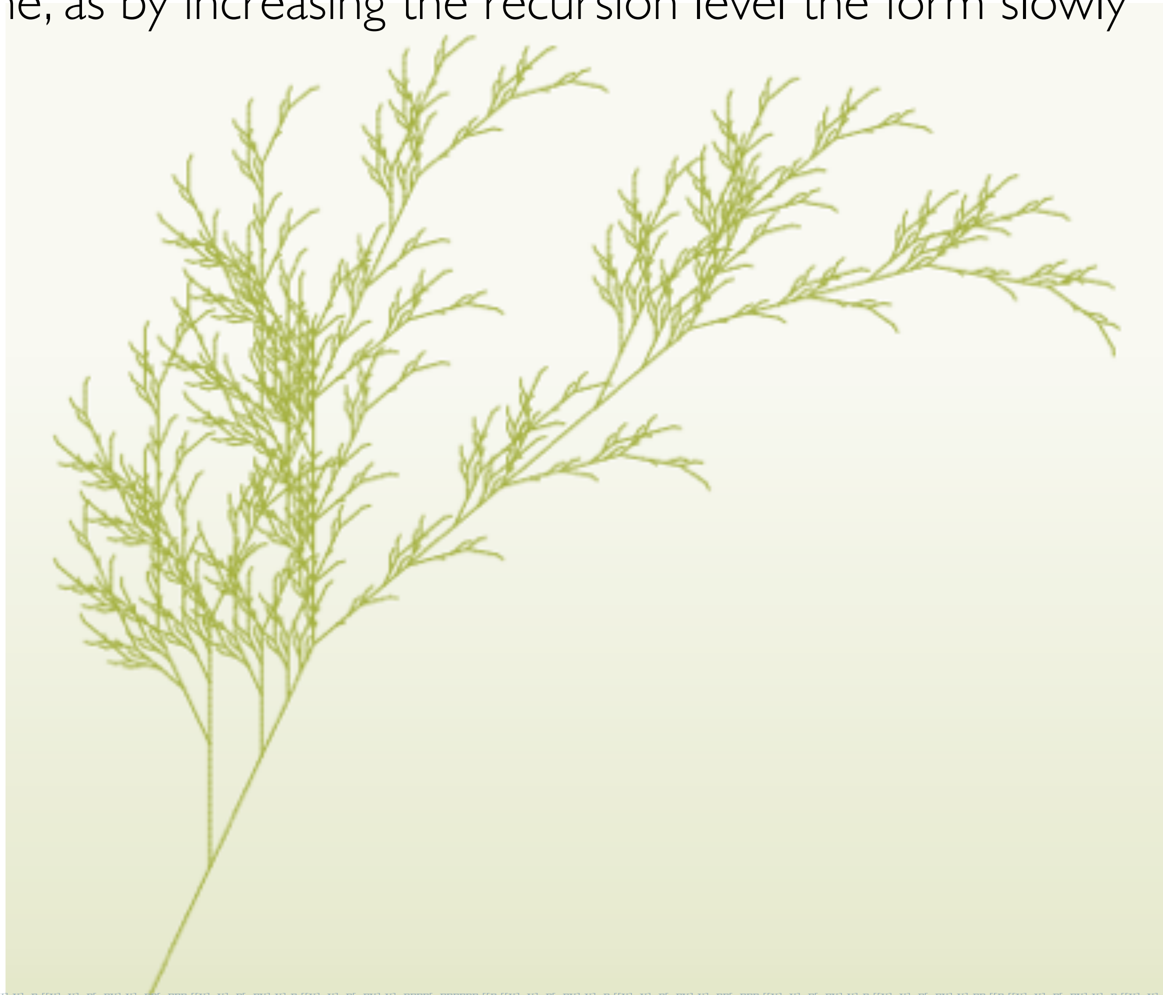
L-system structure

Originally the L-systems were devised to provide a formal description of the development of such simple multi-cellular organisms, and to illustrate the neighborhood relationships between plant cells. Later on, this system was extended to describe higher plants and complex branching structures. The **recursive nature** of the L-**system** rules leads to self-similarity and thereby **fractal-like** forms which are easy to describe with an L-system. Plant models and natural-looking organic forms are similarly easy to define, as by increasing the recursion level the form slowly 'grows' and becomes more complex.

variables : X F
constants : + −
start : X
ruleX : F-[[X]+X]+F[+FX]-X
ruleF : FF
angle : 25°

F : draw forward
+ : turn + 25°
− : turn - 25°

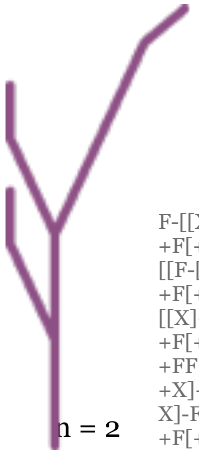




```
F-[[X]+X]+F[+FX]-XFF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-XFFFF-[[FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFF[+FFFFFF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]-FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-
[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-XFFFFFFFF-[[FFFF-[[FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFF[+FFFFFF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]-FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFF-[[FF-[[F-[[X]+X]
+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFF[+FFFFFF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]-FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFFFFFF[+FFFFFFFFFFFF-[[FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]
+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFF[+FFFFFF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]-FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]-FFFF-[[FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]
+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFF[+FFFFFF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]-FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-XFFFFFFFFFFFFFFFF-[[FFFFFFFF-[[FFFF-[[FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-
[[X]+X]+F[+FX]-X]+FFFF[+FFFFFF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]-FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFF-[[FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFF[+FFFFFF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]
+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]-FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFFFFFF[+FFFFFFFFFFFF-[[FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFF[+FFFFFF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]
+F[+FX]-X]-FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]-FFFF-[[FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFF[+FFFFFF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]-FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]
+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFFFFFF-[[FFFF-[[FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFF[+FFFFFF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]-FF-[[F-[[X]+X]+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X]+X]+F[+FX]-X]+FFFF-[[FF-[[F-[[X]+X]
+F[+FX]-X]+F-[[X]+X]+F[+FX]-X]+FF[+FFF-[[X]+X]+F[+FX]-X]-F-[[X1488 curves added to selection.
```

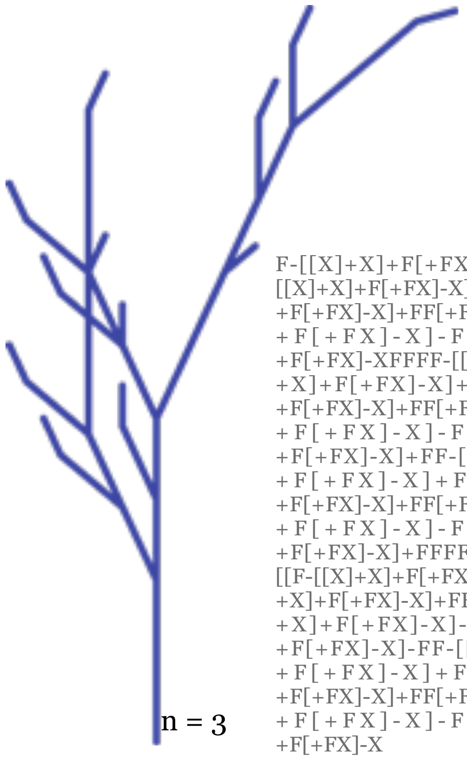
http://en.wikipedia.org/wiki/L-system

L-system structure



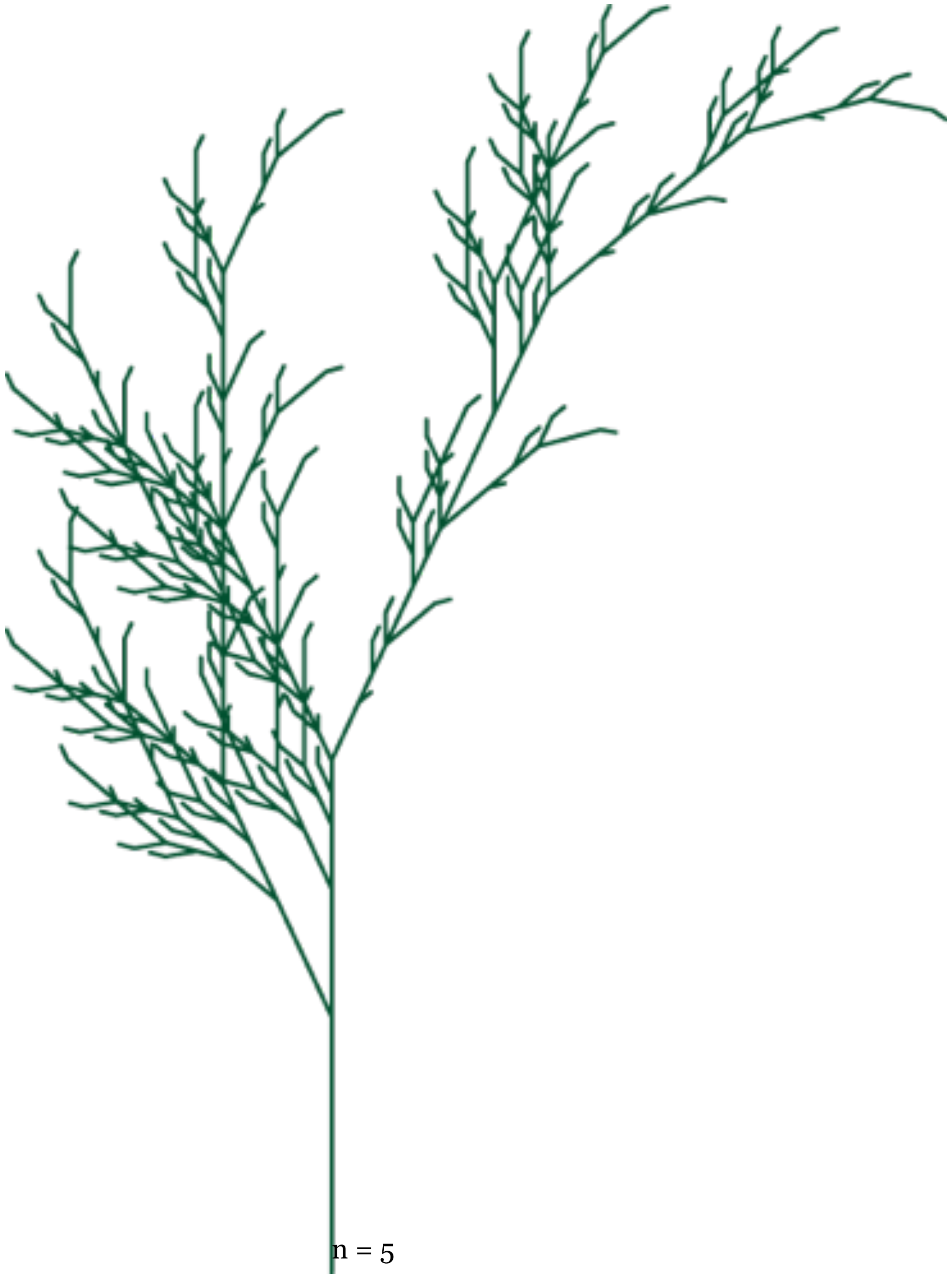
n = 2

```
F-[[X]+X]
+F[+FX]-XFF-
[[F-[[X]+X]
+F[+FX]-X]+F-
[[X]+X]
+F[+FX]-X]
+FF[+FFF-[[X]
+X]+F[+FX]-
X]-F-[[X]+X]
+F[+FX]-X
```



n = 3

```
F-[[X]+X]+F[+FX]-XFF-[[F-
[[X]+X]+F[+FX]-X]+F-[[X]+X]
+F[+FX]-X]+FF[+FFF-[[X]+X]
+F[+FX]-X]-F-[[X]+X]
+F[+FX]-XFFFF-[[FF-[[F-[[X]
+X]+F[+FX]-X]+F-[[X]+X]
+F[+FX]-X]+FF[+FFF-[[X]+X]
+F[+FX]-X]-F-[[X]+X]
+F[+FX]-X]+FF-[[F-[[X]+X]
+F[+FX]-X]+F-[[X]+X]
+F[+FX]-X]+FF[+FFF-[[X]+X]
+F[+FX]-X]-F-[[X]+X]
+F[+FX]-X]+FFFF[+FFFFFF-
[[F-[[X]+X]+F[+FX]-X]+F-[[X]
+X]+F[+FX]-X]+FF[+FFF-[[X]
+X]+F[+FX]-X]-F-[[X]+X]
+F[+FX]-X]-FF-[[F-[[X]+X]
+F[+FX]-X]+F-[[X]+X]
+F[+FX]-X]+FF[+FFF-[[X]+X]
+F[+FX]-X]-F-[[X]+X]
+F[+FX]-X
```



n = 5

L-system algorithm

Step 1 : Formatting string

For all characters of the given string:

```
Loop :
    Check  string character:
        Case “X
            replace with the Rule_X
        Case “F”
            replace with the Rule_F
        Others
            do nothing
End loop
```

Step 2: Generation

Iterate through all characters:

```
Loop :
    Check  string character:
        Case “F”
            Draw a line forward;
        Case “+”
            make a turn(+angle)
        Case “–”
            make a turn(-angle)
        Case “[”
            Stack current conditions
        Case “]”
            Release last conditions
End loop
```