



# TPC2 MSSN

## Relatório

Docente:

Paulo Vieira

Alunos:

Diogo Fernandes nº 39205

Miguel Carreira nº 41543

## Exercício 1 - Compreender e utilizar o código feito em aula

De modo a compreender de melhor maneira o código desenvolvido em aula, efetuamos algumas simulações, tanto a nível do funcionamento da ferramenta Processing como da manipulação da String que dá forma aos nossos Lyndenmayer Systems. Dos sistemas testados, os seguintes deram os resultados que consideramos mais interessantes:

### **Exemplo 1:**

Variáveis: 'F'

Axioma: "F"

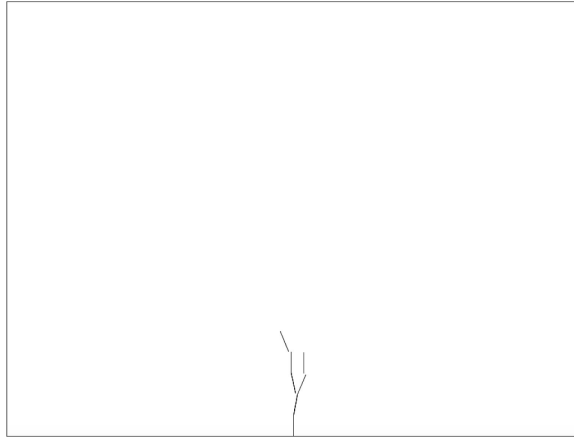
Regras:  $F : F[+[F]-fF]$

Ângulo:  $10^\circ$

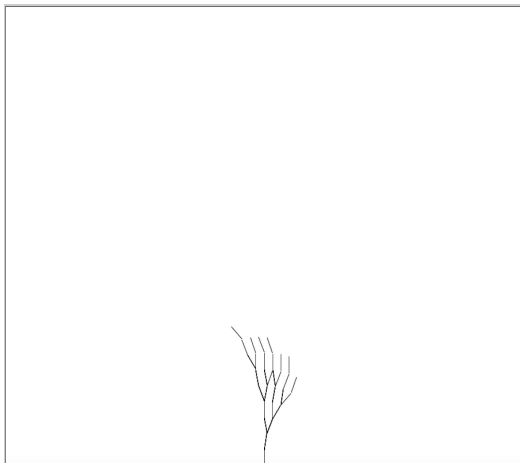
Para interações = 0:



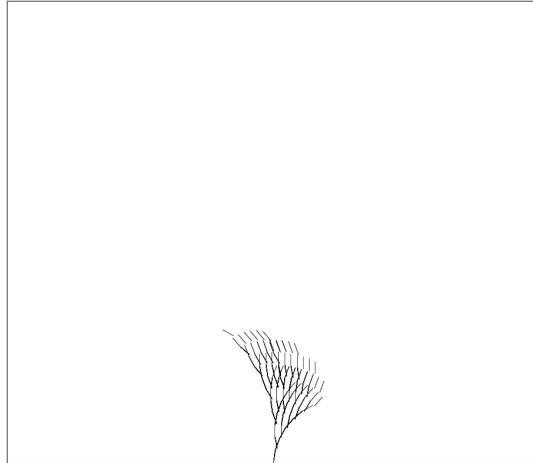
Para interações = 2:



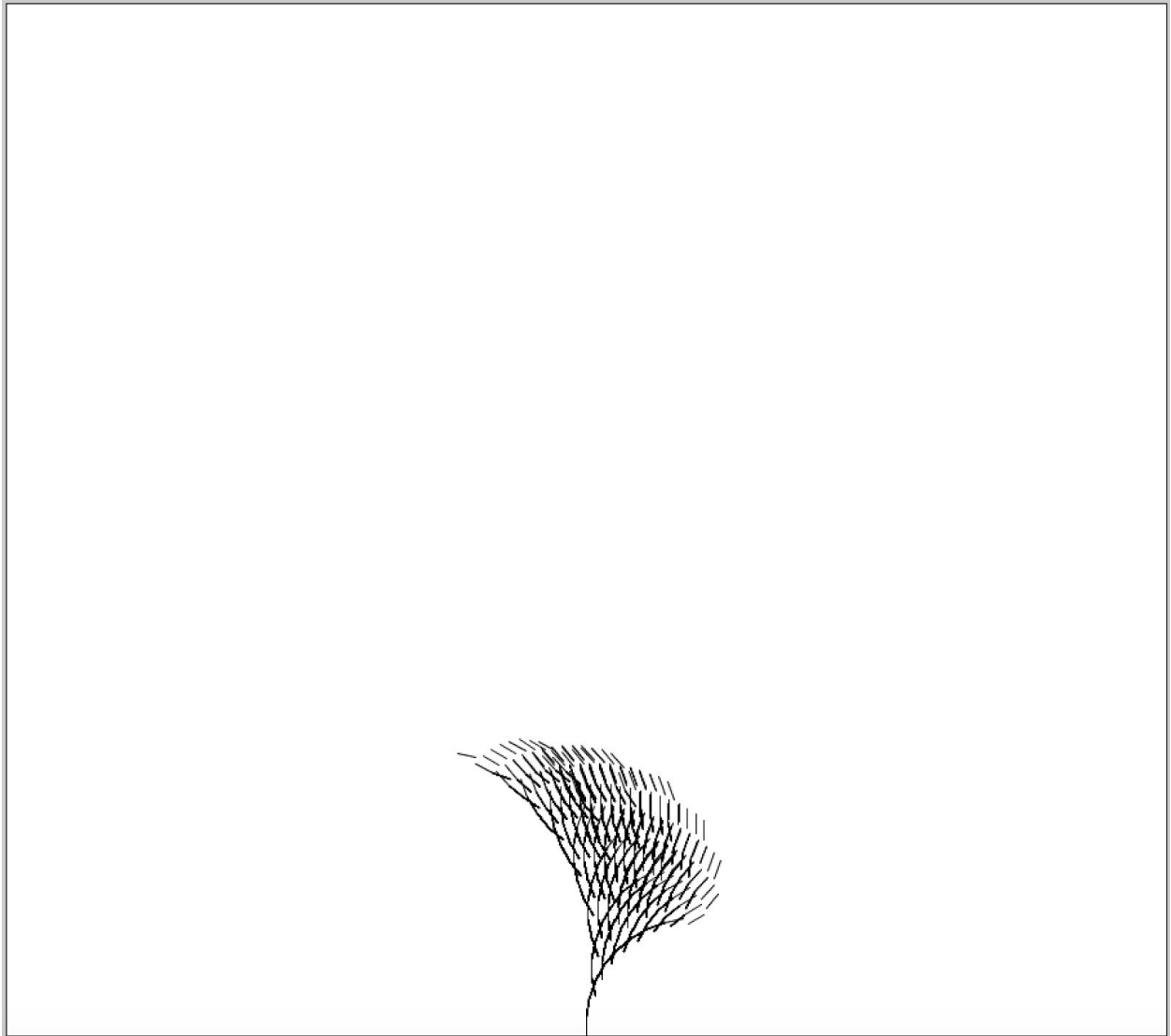
Para interações = 4:



Para interações = 6:



Para interações = 8:



## **Exemplo 2:**

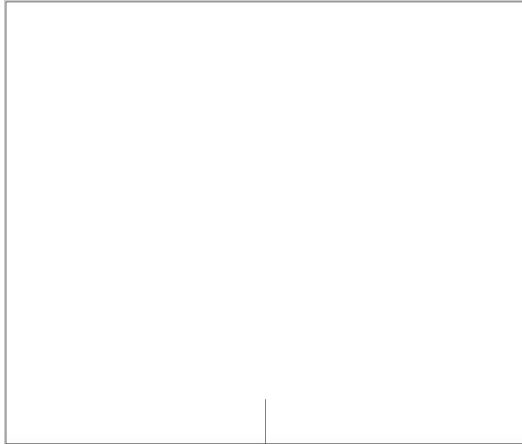
Variáveis: 'F'

Axioma: "F"

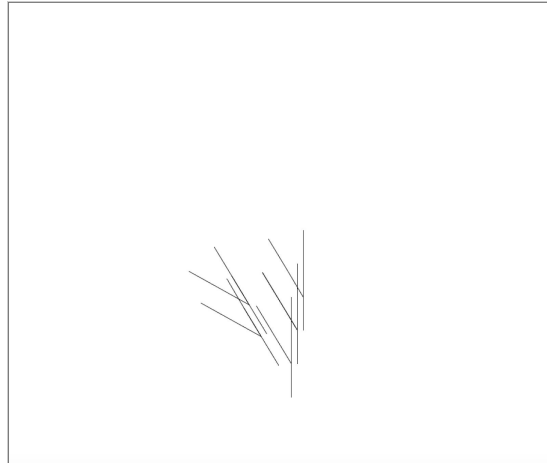
Regras:  $F : f[F-[[FF]+[FF]]+F]$

Ângulo:  $30^\circ$

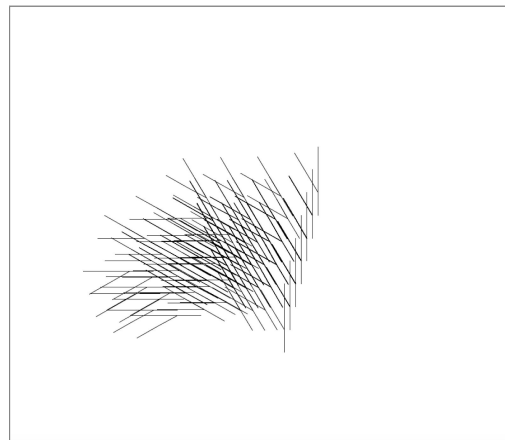
Para interações = 0:



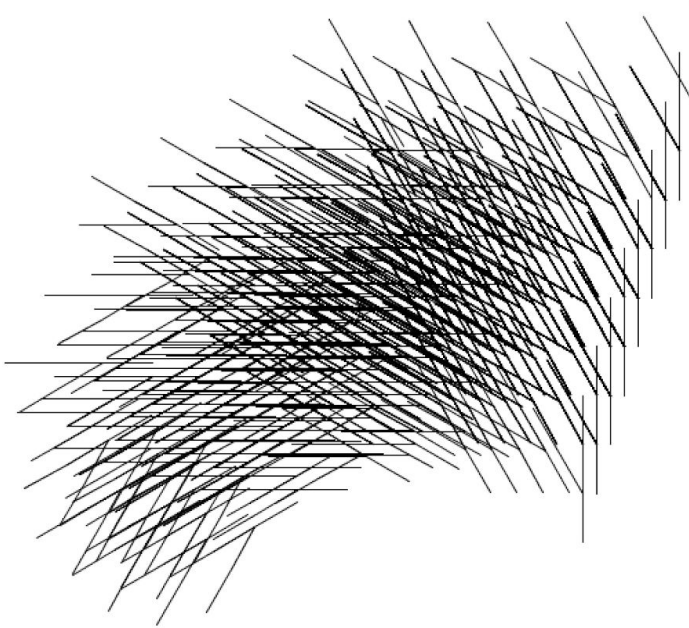
Para interações = 2:



Para interações = 4:



Para interações = 6:



## Exercício 2 - Criação de árvore de frutos

Crie uma árvore de frutos usando como base o seguinte sistema de Lindenmayer com duas regras. Variáveis e Constantes: F, G, +, -, [, ] Axioma: F

Regras:

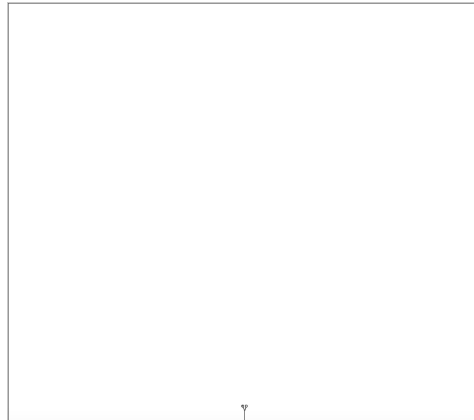
1.  $F \rightarrow G[+F]-F$

2.  $G \rightarrow GG$

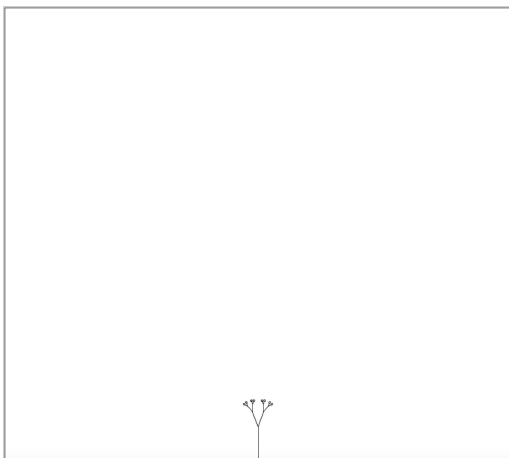
Para iterações = 0:



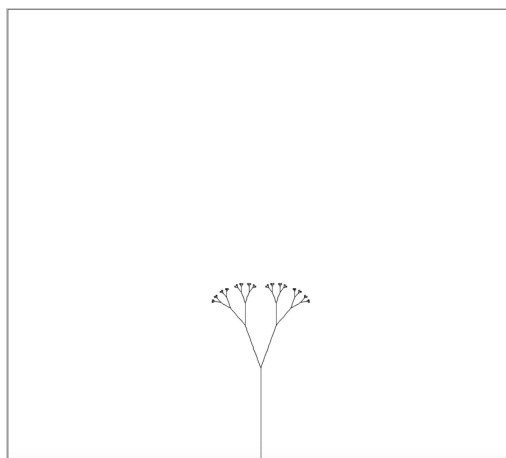
Para iterações = 2:



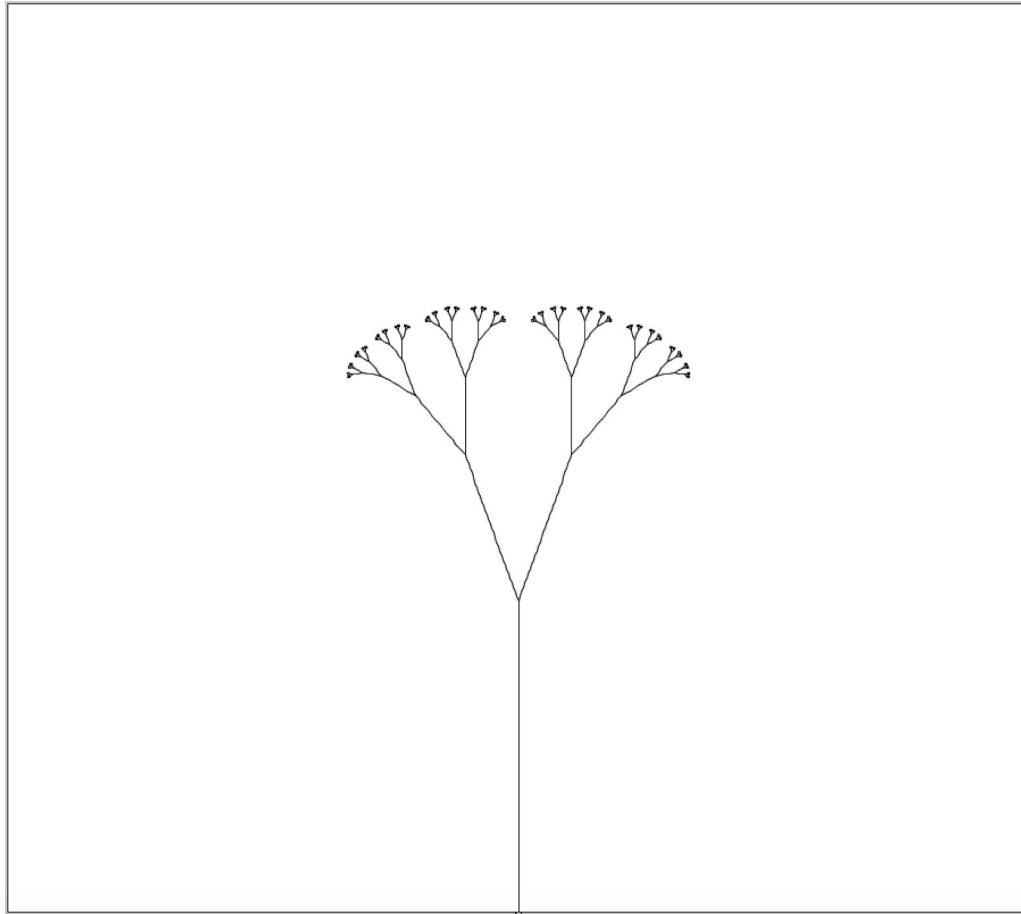
Para iterações = 4:



Para iterações = 6:



Para iterações = 7:



Este sistema implementa uma segunda regra, envolvendo uma nova variável 'G', que permitiu a criação de um sistema semelhante ao crescimento de uma árvore, que desenvolve "frutos" nas extremidades dos ramos.

## Exercício 3 - Curva de Koch

1. Construa algumas variantes da curva de Koch (e.g., veja [aqui](#)) e indique qual a dimensão fractal das curvas produzidas
2. Reproduza uma variante da curva de Koch usando um sistema com regras estocáticas, onde a probabilidade de a curva virar para cima ou para baixo é de 50%

3.1

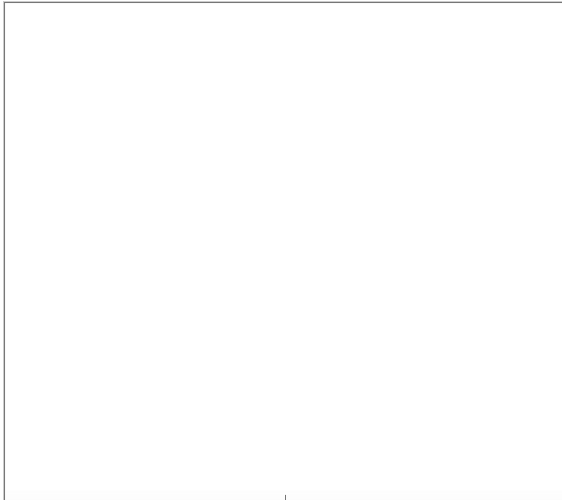
Curva de Koch:

Variáveis: 'F'

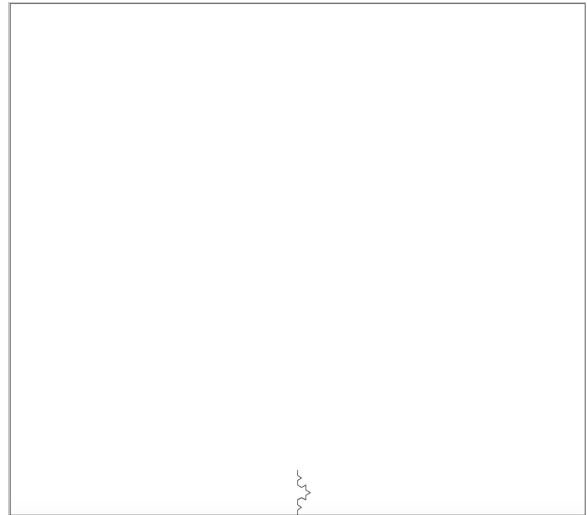
Regras:  $F : F+F--F+F$

Axioma: F

Para iterações = 0:

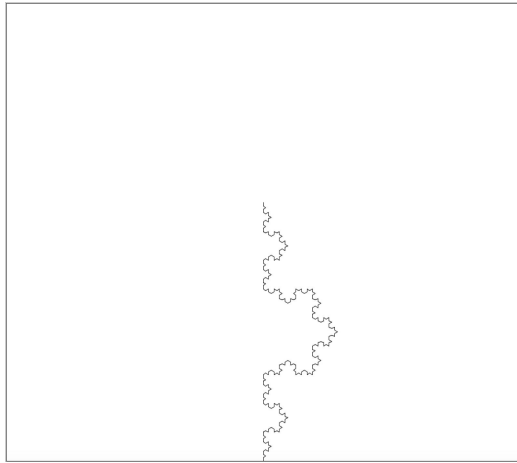


Para iterações = 2:

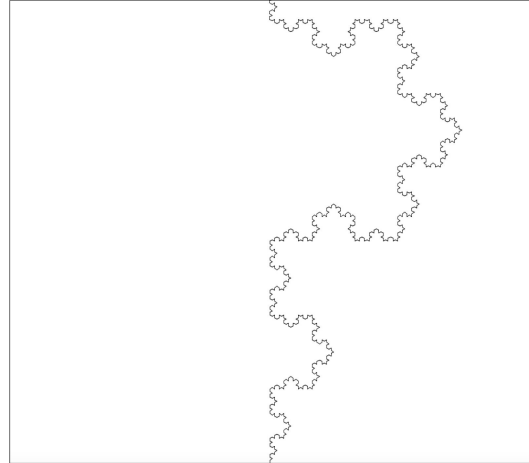




Para iterações = 4:



Para iterações = 5:



Sabemos que a dimensão 'd' da curva de Koch é calculada pela seguinte equação:

$$d = \frac{\log P}{\log S}$$

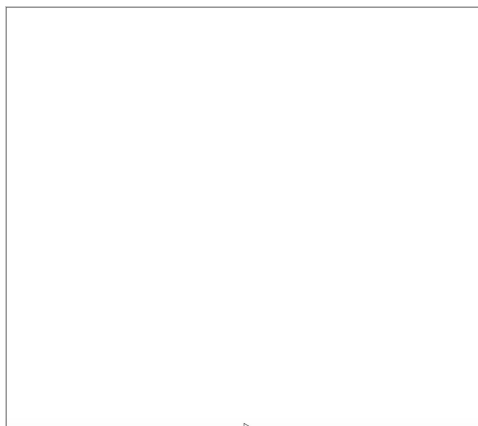
Sabemos também que P representa o numero de linhas que cada linha 'gera' na nova iteração, neste caso P=4, e S=3, pelo que,

$$\frac{\log 4}{\log 3} \approx 1.26185...$$

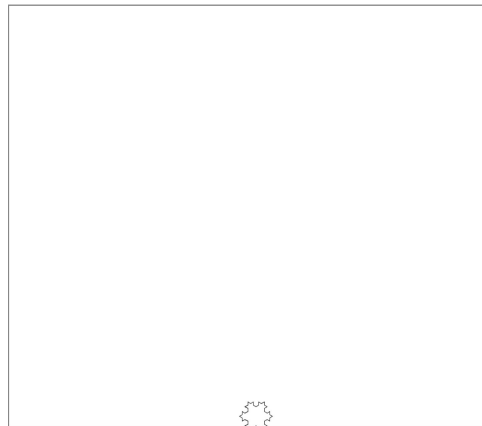
Snowflake de Koch

Partindo da regra anterior, se for aplicada num triângulo como axioma ("F--F--F").

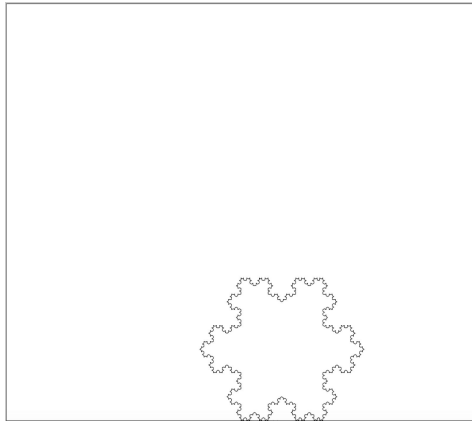
Para iterações = 0:



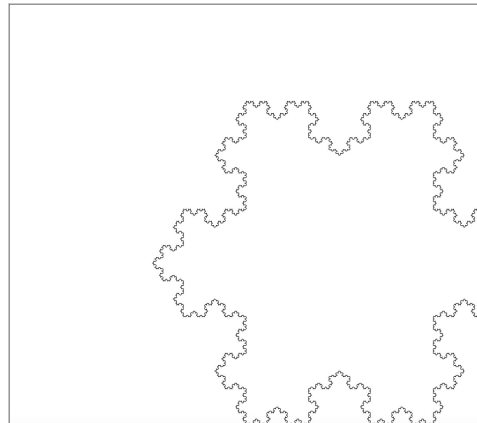
Para iterações = 2:



Para iterações = 4:



Para iterações = 5:

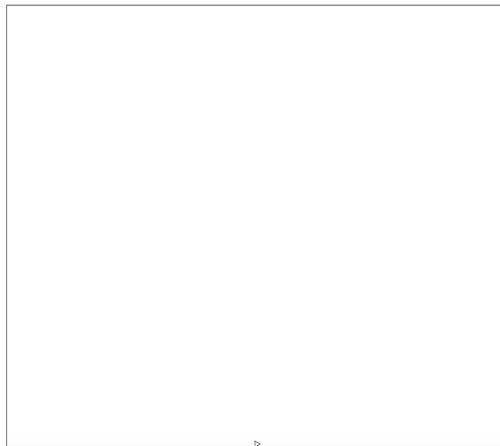


### 3.2. E Exercício 4:

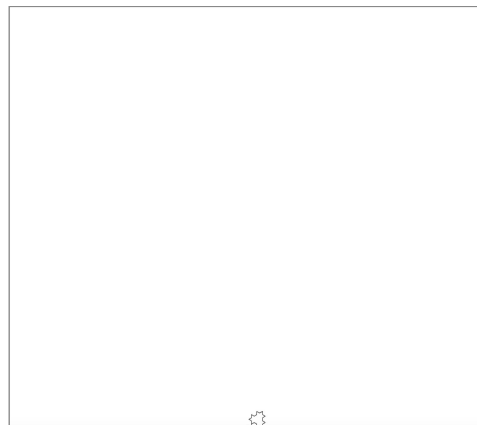
Para esta nova implementação, temos a classe `StochasticSystem`, que permite que seja indicado um grau de incerteza na criação da nova geração, mantendo as regras implementadas anteriormente mas adicionando um elemento aleatório.

Com um índice de incerteza de 50%:

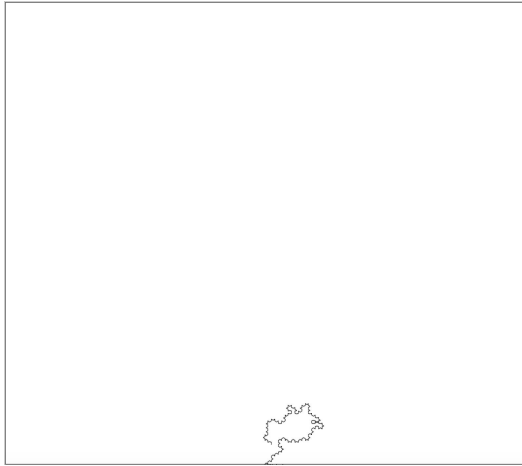
Para iterações = 0:



Para iterações = 2:



Para iterações = 4:



Para iterações = 6:

