Trabalho Prático - Matemática Discreta

Nome: Letícia Scofield Lenzoni

1.

Os códigos dos fractais estão nos arquivos “fractali.c”, “fractalii.c” e “fractaliii.c”, respectivamente. Para a execução dos códigos, consulte o arquivo “leiame.txt”.

(i) Floco de neve onda senoidal 2 de von Koch
(ii) Preenchimento de espaço de Hilbert
(iii) Fractal Letícia Scofield

2..

A abordagem adotada por mim foi uma versão iterativa usando vetor de char para gerar os fractais. Nessa abordagem, começa-se com um estágio inicial (axioma) e aplica a regra iterativamente para gerar o próximo estágio, até alcançar o estágio desejado. Os caracteres de cada estágio são armazenados em um vetor de char (também em um arquivo para guardar todos os estágios) e lidos posteriormente para gerar o próximo estágio.

Existem outras abordagens possíveis para implementar fractais, como a versão iterativa com arquivos e recursiva mencionada. Na abordagem recursiva, em vez de iterar explicitamente por cada estágio, você define uma função recursiva que chama a si mesma para gerar os caracteres do estágio desejado. A recursão ocorre até que uma condição de parada seja alcançada.

As três as estratégias têm pontos positivos e negativos:

Versão Iterativa com Vetor de Char:

Pontos Positivos:

- O uso de um vetor de char como armazenamento intermediário é mais eficiente em termos de velocidade de acesso aos dados do que o uso de arquivos.
- Não há custo adicional de armazenamento em disco, tornando essa abordagem mais eficiente em termos de uso de recursos.

Pontos Negativos:

- Pode haver restrições de memória se o fractal for muito grande.
- A manipulação pode ser mais complexa do que o uso de arquivos, especialmente para substituir os caracteres corretos a cada iteração.

Versão Iterativa com Arquivos:

Pontos Positivos:

- Permite armazenar os estágios intermediários, o que pode ser útil para análise ou visualização posterior.
- O uso de arquivos pode facilitar a manipulação e substituição dos caracteres corretos em cada iteração.

Pontos Negativos:

- Pode ser menos eficiente em termos de velocidade de acesso aos dados do que o uso de um vetor de char.
- Pode haver um custo adicional de armazenamento em disco, especialmente se os fractais forem complexos e exigirem muitos estágios.
- O processo de gravação e leitura dos arquivos pode ser mais lento do que a manipulação direta de um vetor de char.

Versão recursiva:

Pontos positivos:

- Geralmente mais concisa, pois a recursão permite uma representação mais direta do processo de geração dos fractais.
- Pode ser mais eficiente em termos de tempo de execução, especialmente para estágios maiores, pois evita a necessidade de armazenar e ler caracteres intermediários.

Pontos negativos:

- Pode exigir um maior entendimento do conceito de recursão e pode ser mais difícil de depurar em alguns casos.
- Pode consumir mais memória se a recursão for muito profunda.

3.

(i) Para o Floco de neve onda senoidal 2 de von Koch, foi feita essa tabela de símbolos:

Floco de neve onda senoidal 2 de von Koch				
	Quantidade "F"	Quantidade "-"	Quantidade "+"	Total de Símbolos
Axioma	1	0	0	1
Estágio 1	8	3	3	14
Estágio 2	64	27	27	118
Estágio 3	512	219	219	950
Estágio 4	4096	1755	1755	7606

Ao analisar o padrão dos símbolos, foi possível fazer essas equações de recorrência:

- Para f(n) sendo a quantidade de “F” no estágio n:

  $f(0) = 1$

  $f(n) = 8f(n-1)$

- Para s(n) sendo a soma das quantidades de “F”,”-” e “+” no estágio n:

  $s(0) = 1$

  $s(n) = 8s(n-1) + 6$

(ii) Para o Preenchimento de espaço de Hilbert, foi feita essa tabela de símbolos:

	Preenchimento de espaço de Hilbert						
	Quantidade "F"	Quantidade "-"	Quantidade "+"	Quantidade "X"	Quantidade "Y"	Total de Símbolos (sem X, Y)	Total de Símbolos (com X, Y)
Axioma	0	0	0	1	0	0	1
Estágio 1	3	2	2	2	2	7	11
Estágio 2	15	10	10	8	8	35	51
Estágio 3	63	42	42	32	32	147	211
Estágio 4	255	170	170	128	128	595	851

Ao analisar o padrão dos símbolos, foi possível fazer essas equações de recorrência:

- Para f(n) sendo a quantidade de "F" no estágio n:

    $f(0) = 0$

    $f(n) = 4f(n-1) + 3$

- Para s(n) sendo a soma das quantidades de "F","-" e "+" no estágio n:

    $s(0) = 0$

    $s(n) = 4s(n-1) + 7$

- Para t(n) sendo a soma das quantidades de "F","-" ,"+","X" e "Y" no estágio n:

    $t(0) = 1$

    $t(n) = 4t(n-1) + 7$

(iii) Para o Fractal Letícia Scofield, foi feita essa tabela de símbolos:

	Fractal Letícia Scofield						
	Quantidade "F"	Quantidade "-"	Quantidade "+"	Quantidade "X"	Quantidade "Y"	Total de Símbolos (sem X, Y)	Total de Símbolos (com X, Y)
Axioma	4	0	4	2	2	8	12
Estágio 1	24	8	12	10	10	44	64
Estágio 2	124	48	52	50	50	224	324
Estágio 3	624	248	252	250	250	1124	1624
Estágio 4	3124	1248	1252	1250	1250	5624	8124

Ao analisar o padrão dos símbolos, foi possível fazer essas equações de recorrência:

- Para f(n) sendo a quantidade de "F" no estágio n:

    $f(0) = 4$

    $f(n) = 5f(n-1) + 4$

- Para s(n) sendo a soma das quantidades de "F","-" e "+" no estágio n:

    $s(0) = 8$

    $s(n) = 5s(n-1) + 4$

- Para s(n) sendo a soma das quantidades de "F","-" ,"+","X" e "Y" no estágio n:

    $t(0) = 12$

    $t(n) = 5t(n-1) + 4$

4.

(i) Floco de neve onda senoidal 2 de von Koch:

- Expandindo os termos da equação f(n) para entender o padrão:

f(0) = 1
f(1) = 8f(0) = 8
f(2) = 8f(1) = 8 * 8 = 64
f(3) = 8f(2) = 8 * 64 = 512
…

Podemos ver que f(n) é igual a 8 elevado a n, ou seja, f(n) = 8^n. Portanto, o custo assintótico dessa equação de recorrência é Θ(8^n).

- Expandindo os termos da equação s(n) para entender o padrão:

  s(0) = 1
  s(1) = 8s(0) + 6 = 8 * 1 + 6 = 14
  s(2) = 8s(1) + 6 = 8 * 14 + 6 = 118
  s(3) = 8s(2) + 6 = 8 * 118 + 6 = 950
  ...

  Podemos ver que s(n) é igual a 8 elevado a n mais 6, ou seja, s(n) = 8^n + 6. Como o termo dominante é 8^n, o custo assintótico dessa equação de recorrência depende dele, sendo então Θ(8^n).

(ii) Preenchimento de espaço de Hilbert:
- Expandindo os termos da equação f(n) para entender o padrão:

  f(0) = 0
  f(1) = 4f(0) + 3 = 4 * 0 + 3 = 3
  f(2) = 4f(1) + 3 = 4 * 3 + 3 = 15
  f(3) = 4f(2) + 3 = 4 * 15 + 3 = 63
  ...

  Podemos ver que f(n) é igual a 4 elevado a n mais 3, ou seja, f(n) = 4^n + 3. Como o termo dominante é 4^n, o custo assintótico dessa equação de recorrência depende dele, sendo então Θ(4^n).

- Expandindo os termos da equação s(n) para entender o padrão:

  s(0) = 0
  s(1) = 4s(0) + 7 = 4 * 0 + 7 = 7
  s(2) = 4s(1) + 7 = 4 * 7 + 7 = 35
  s(3) = 4s(2) + 7 = 4 * 35 + 7 = 147
  ...

  Podemos ver que s(n) é igual a 4 elevado a n mais 7, ou seja, s(n) = 4^n + 7. Como o termo dominante é 4^n, o custo assintótico dessa equação de recorrência depende dele, sendo então Θ(4^n).
  O mesmo ocorre para t(n), sendo também Θ(4^n).

(iii) Fractal Letícia Scofield

- Expandindo os termos da equação f(n) para entender o padrão:

  $f(0) = 4$
  $f(1) = 5f(0) + 4 = 5 * 4 + 4 = 24$
  $f(2) = 5f(1) + 4 = 5 * 24 + 4 = 124$
  $f(3) = 5f(2) + 4 = 5 * 124 + 4 = 624$
  ...

  Podemos ver que t(n) é igual a 5 elevado a n mais 4, ou seja, t(n) = 5^n + 4. Como o termo dominante é 5^n, o custo assintótico dessa equação de recorrência depende dele, sendo então Θ(5^n).
  O mesmo ocorre para s(n) e t(n), sendo também Θ(5^n).

5.

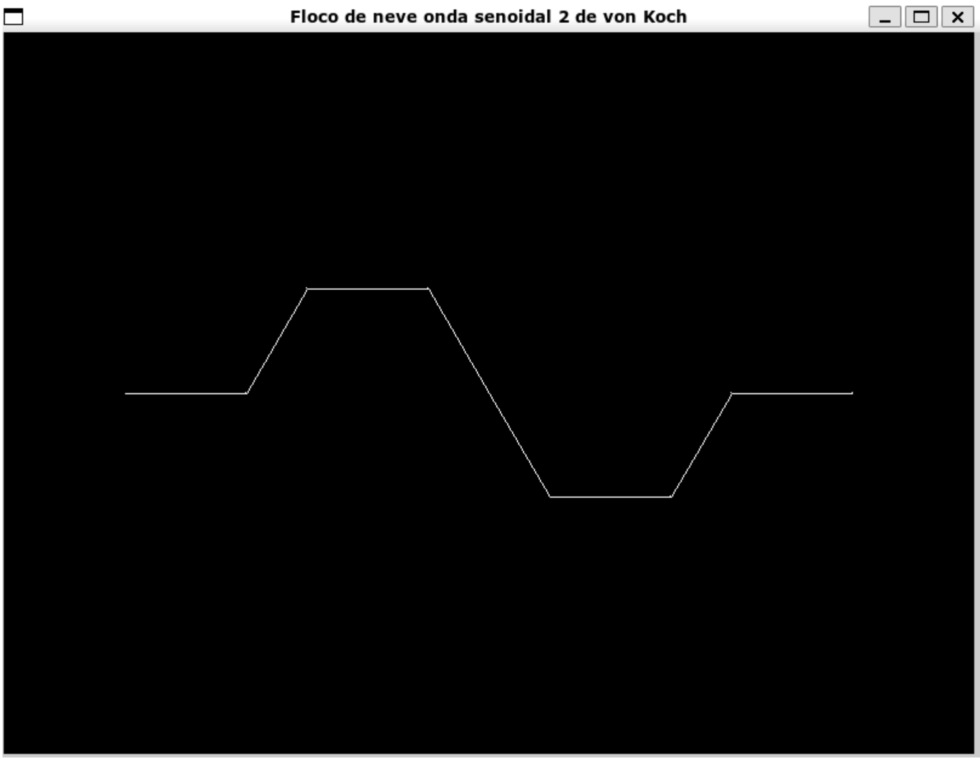
As imagens dos três fractais foram criadas utilizando a biblioteca SDL2, a qual também está presente no código de cada um deles. Optei por utilizar essa biblioteca para desenvolver meus próprios programas de desenho de fractais, devido aos seus recursos gráficos avançados, facilidade de uso e à ampla quantidade de conteúdo online disponível para aprendizado. Além disso, ao ter escrito o código para a criação das imagens, adquiri um melhor entendimento do processo envolvido na geração dos fractais.

As linhas de cada fractal foram reduzidas a cada estágio, para que as imagens ocupassem um tamanho similar na tela.

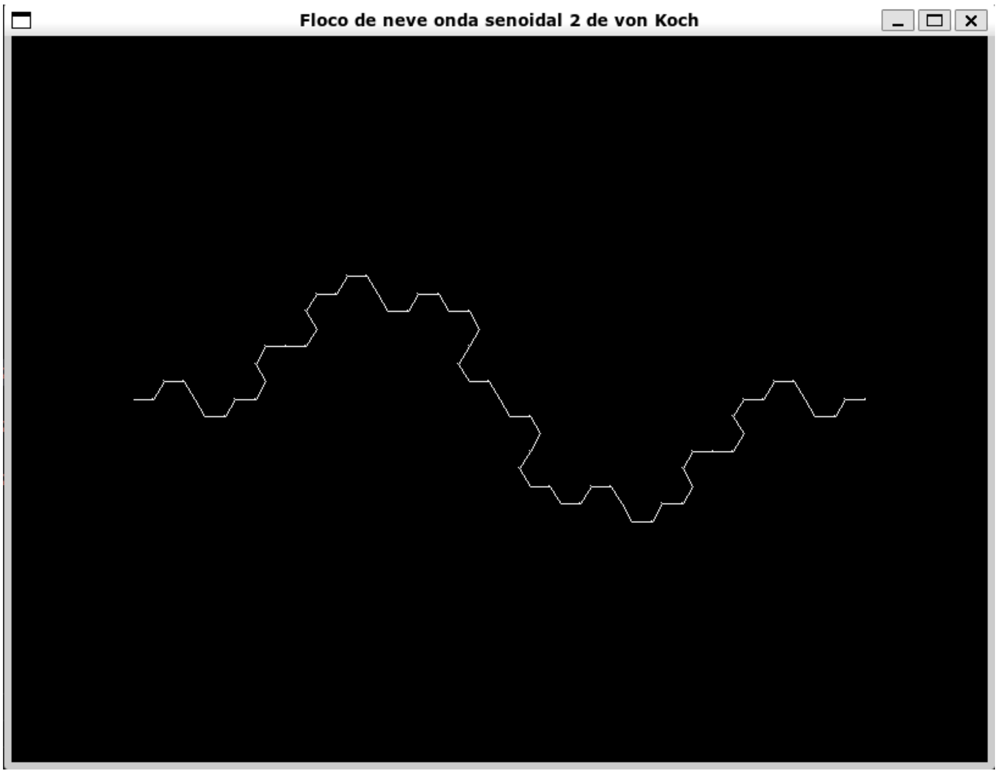
(i) Floco de neve onda senoidal 2 de von Koch

- Axioma: F
- Ângulo: 60
- Regra: F-F+F+FF-F-F+F

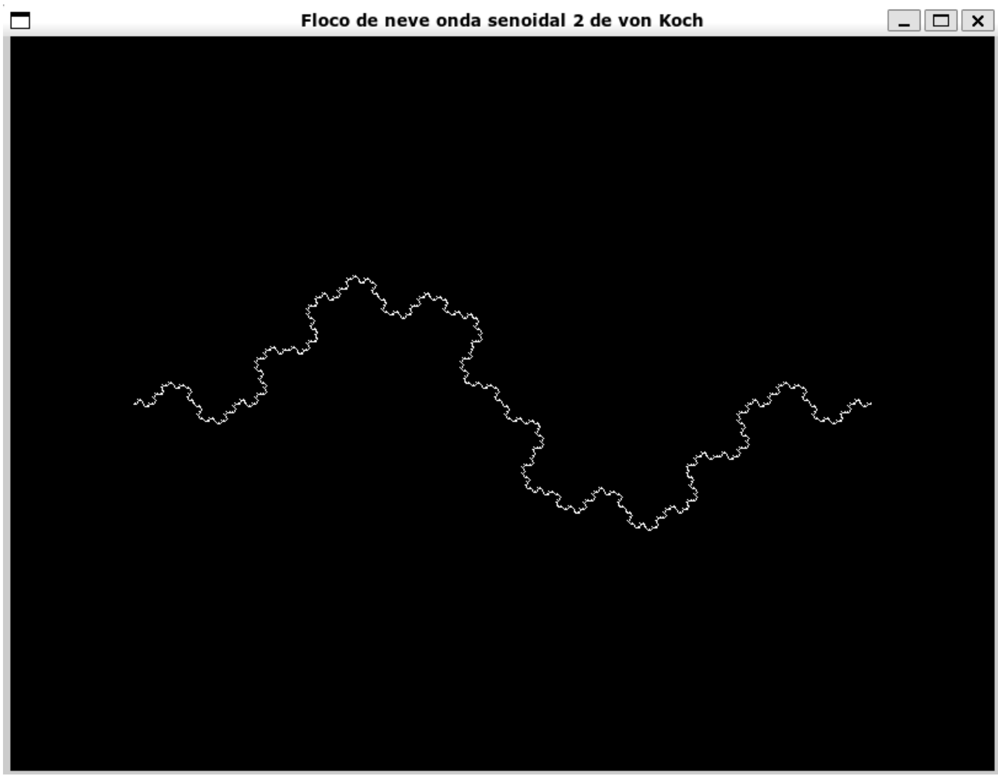
Estágio 1:



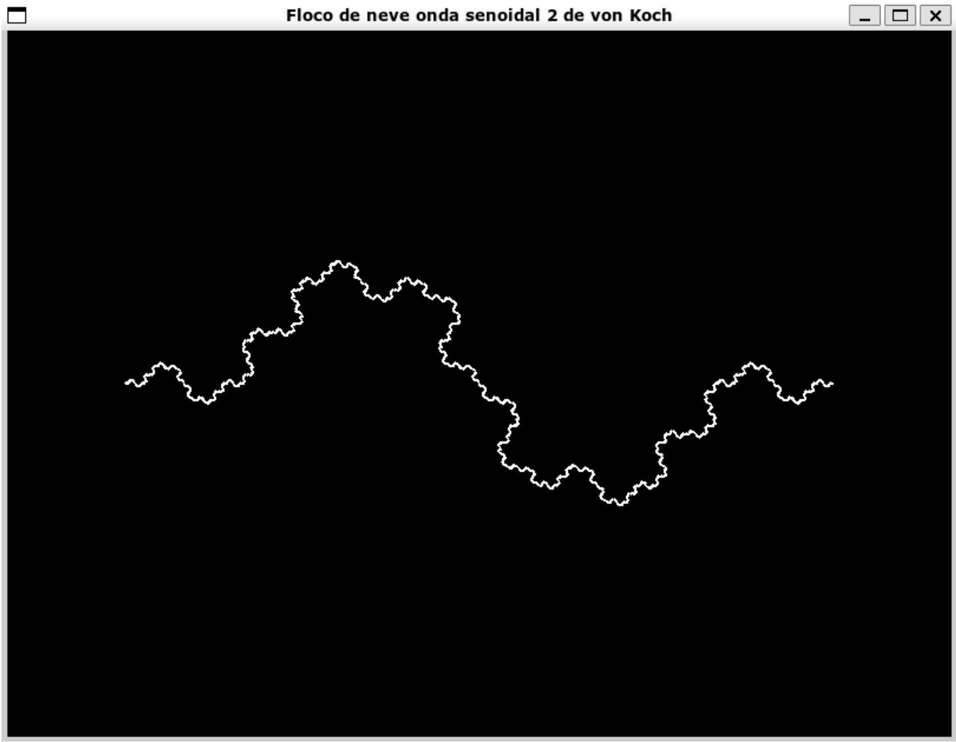
Estágio 2:



Estágio 3:



Estágio 4

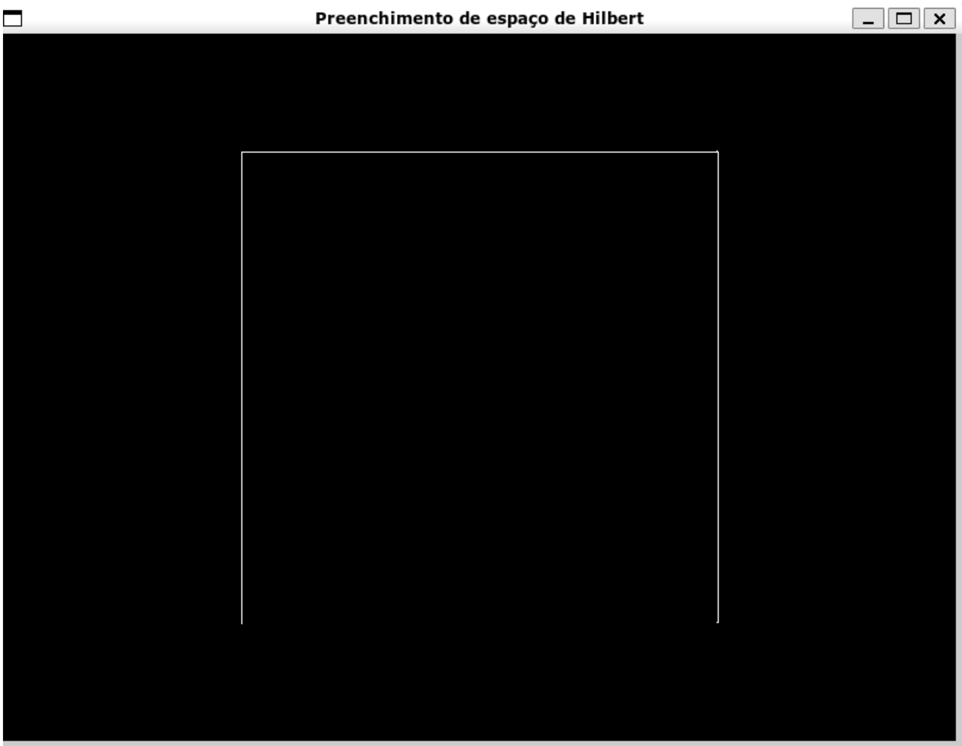


:

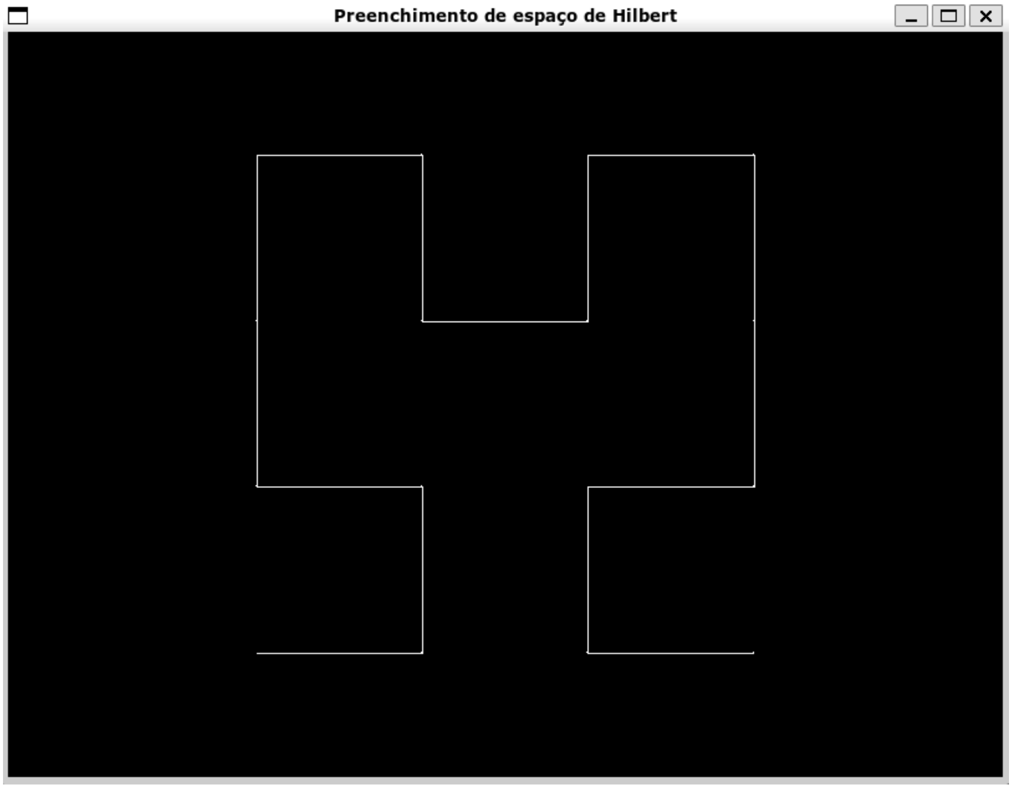
(ii) Preenchimento de espaço de Hilbert

- Axioma: X
- Ângulo: 90
- Regra de X: -YF+XFX+FY-
- Regra de Y: +XF-YFY-FX+

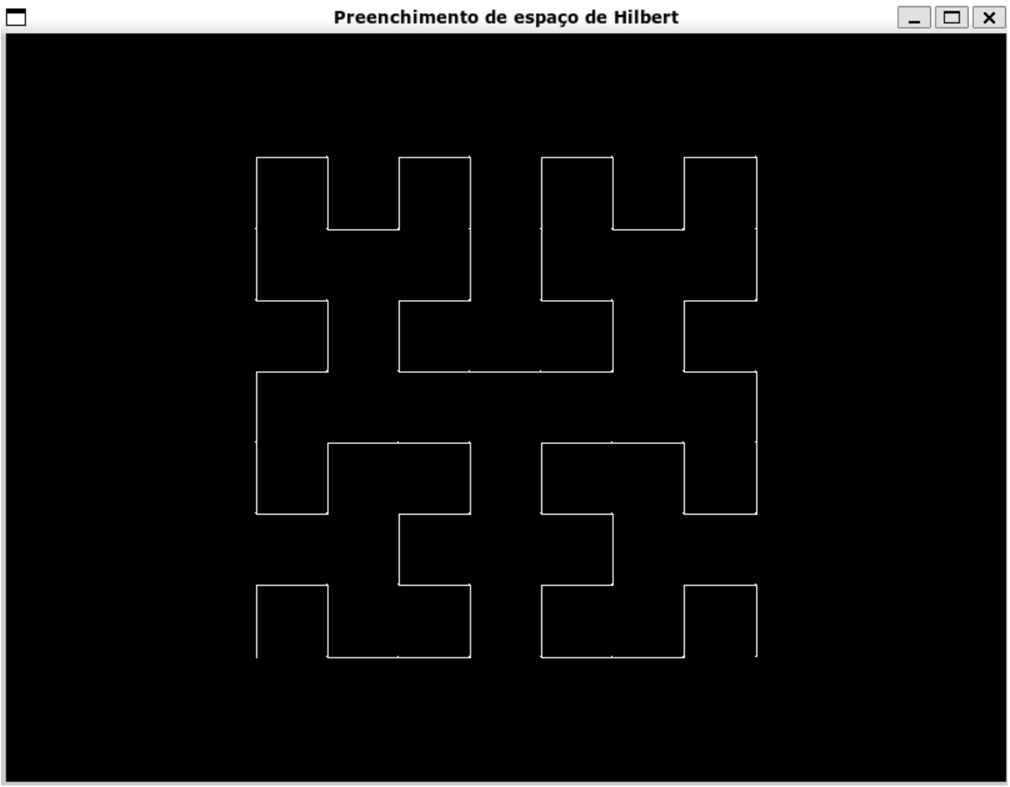
Estágio 1:



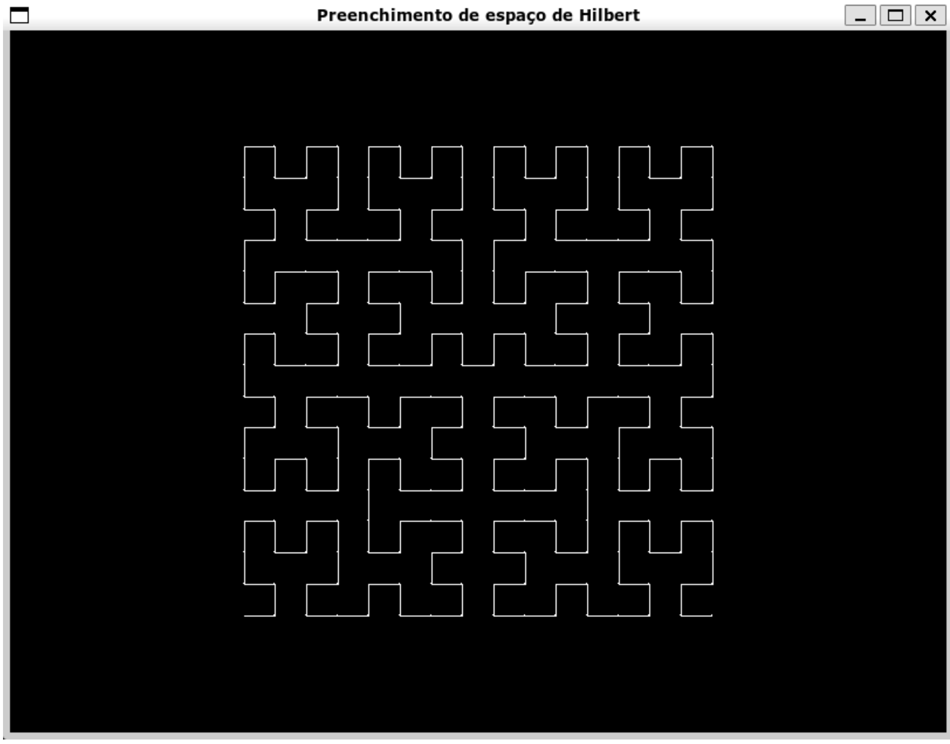
Estágio 2:



Estágio 3:



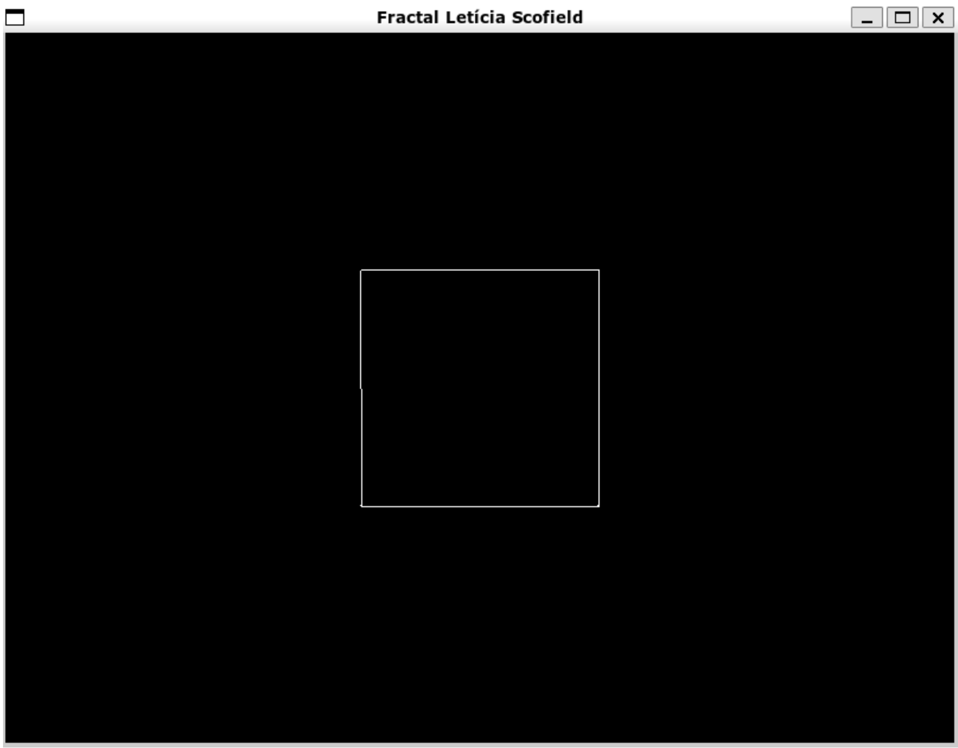
Estágio 4:



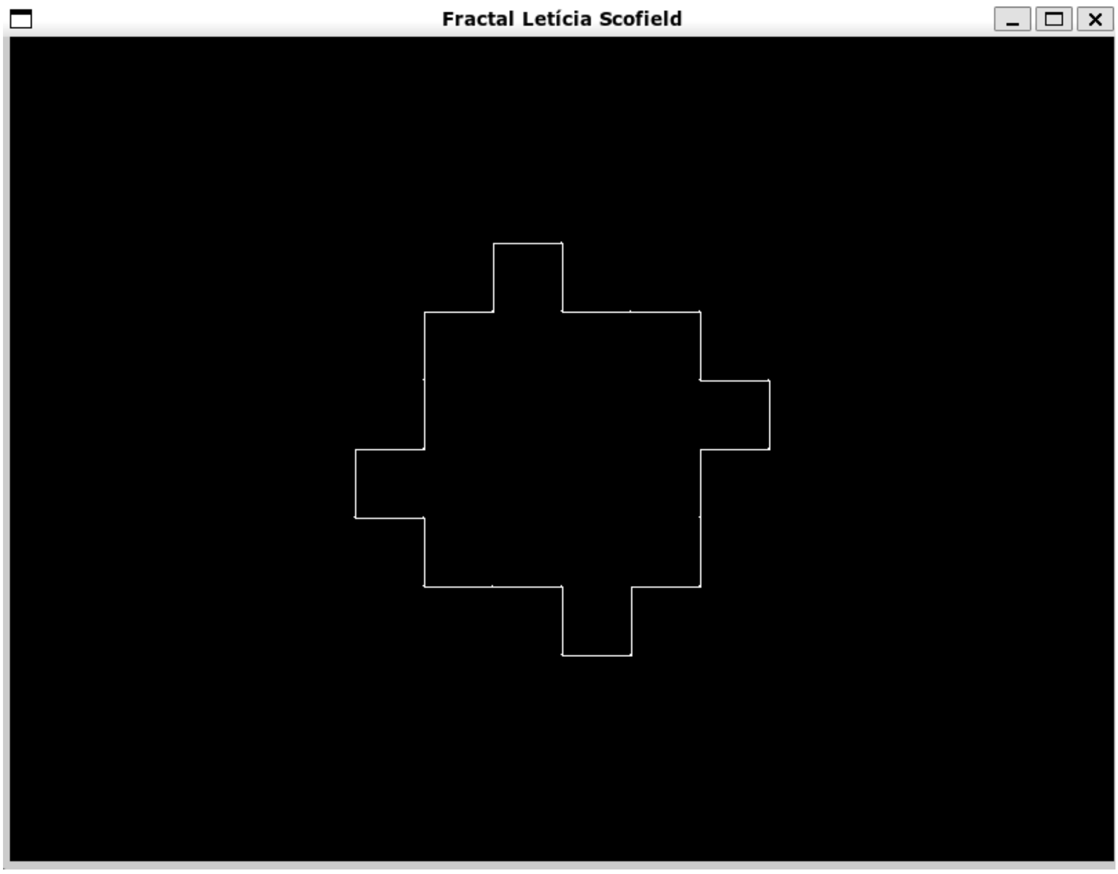
(iii) Fractal Letícia Scofield

- Axioma: XF+YF+XF+YF+
- Ângulo: 90
- Regra de X: XF-XF+XF+XF-XF
- Regra de Y: YF-YF+YF+YF-YF

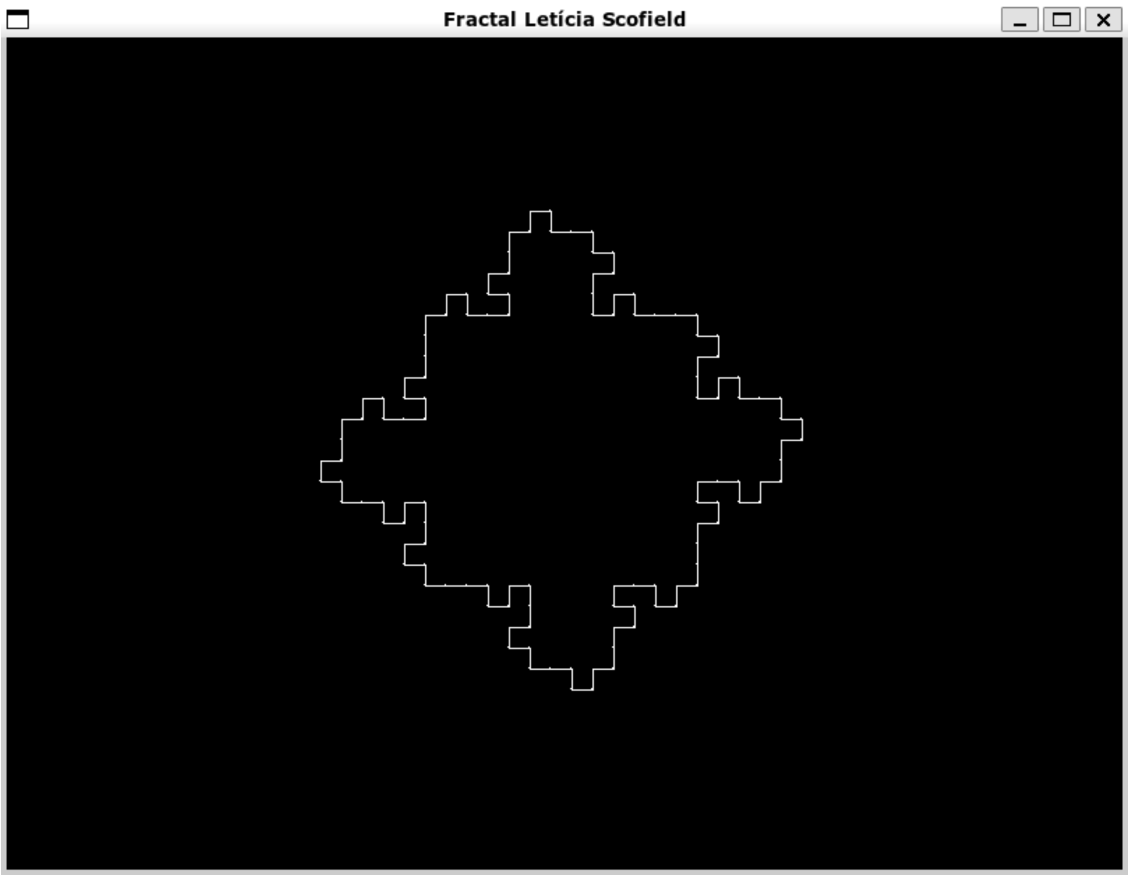
Estágio 0:



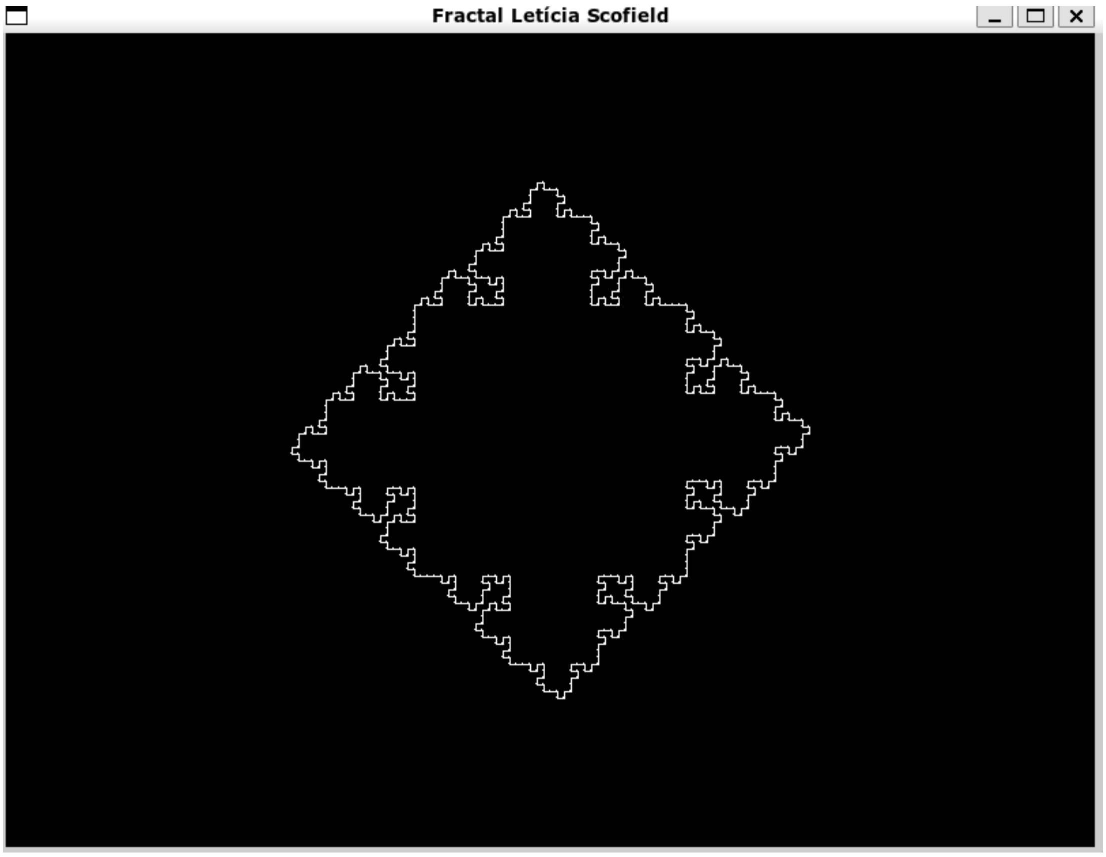
Estágio 1:



Estágio 2:



Estágio 3:



Estágio 4: