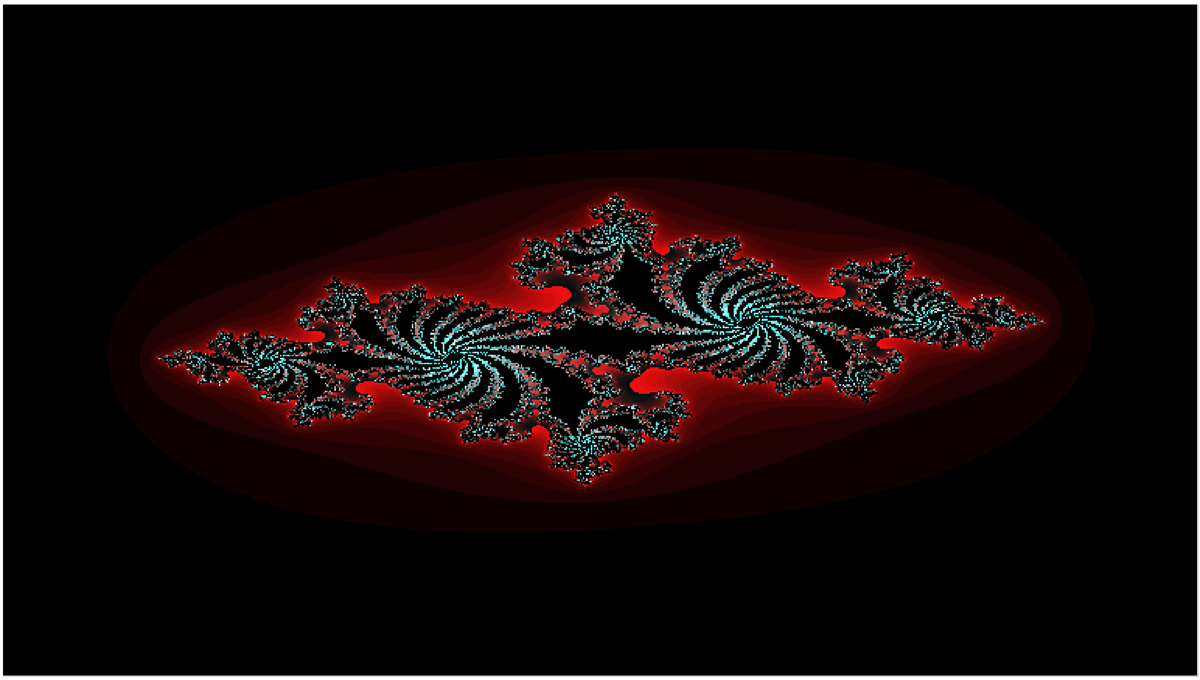
Instituto Superior de Engenharia de Lisboa

Licenciatura em Engenharia Informática e Multimédia

Modelação e Simulação de Sistemas Naturais - 2022/2023 SI

Projeto 3



Docente Paulo Vieira

Realizado por :
Pedro Silva 48965

21 de dezembro de 2022

Conteúdo

Lista de Figuras

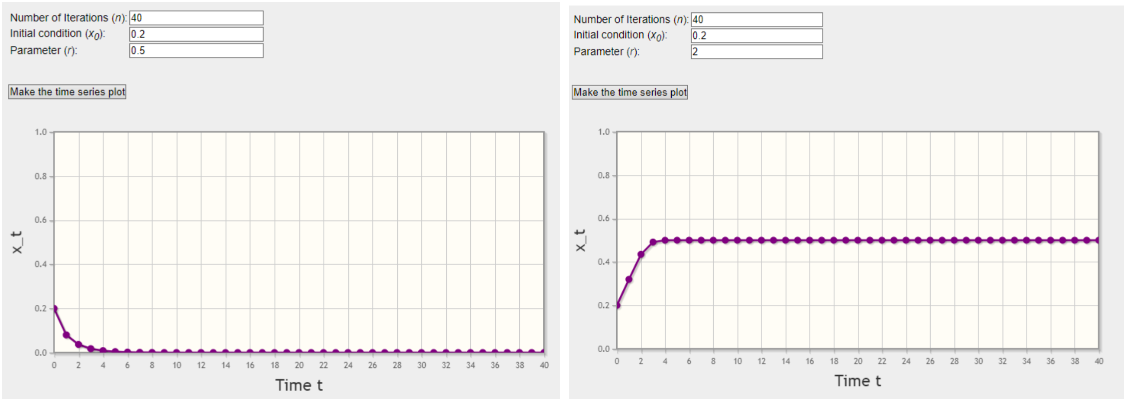
1 Introdução

Este projeto pretende consolidar e avaliar os conhecimentos adquiridos na terceira parte da disciplina Modelação e Simulação de Sistemas Naturais tendo como objetivo o estudo de uma Função Logística em ..., o desenvolvimento do Jogo do Caos e a criação de Gramáticas de Lindenmayer e de conjuntos de Julia e Mandelbrot. O primeiro objetivo deste trabalho era, para a função "f(x) = rx(1-x)", estudar as trajetórias (órbitas, sequências temporais) que resultam dos diversos valores de r possíveis. Com o Jogo do Caos foi primeiro pedido que criássemos o jogo com as regras normais para três pontos fixos, ou seja, um triângulo. De seguida era facultativo fazer com que o utilizador possa determinar a posição de cada ponto fixo. Para as Gramáticas de Lindenmayer foi necessário criar pelo menos dois objetos fractais usando L-Systems. Também foi requisitado a criação de uma árvore de frutos usando o conjunto de regras presente no enunciado. Tendo por base o código desenvolvido em aula foi necessário exprimentar formas alternativas de colorir os pontos que não pertencem ao conjunto de Mandelbrot. Também foi pedido para adicionar a esta ferramenta uma janela em que nesta seja visualizado um determinado conjunto de Julia determinado pelo utilizador ao este escolher um ponto na área que contém o conjunto de Mandelbrot. Por fim, de forma facultativa, foi pedido para adicionar ainda mais funcionalidades a esta ferramenta, pelo que, nós adicionámos a possibilidade de controlar o conjuto de Julia através do arraste do rato e também uma forma de o conjunto esteja sempre a mudar, ao utilizar funções seno e cosseno.

2 A. Função Logística

O objetivo da primeira parte do trabalho era, com a ajuda do programa Hornacek de David P. Feldman estudar a função logística, f(x) = rx(1-x). Para os diferentes valores do parâmetro r foi pedido para encontrar os que resultem em trajetórias que convirjam para:

1. Ponto fixo igual a zero;
2. Ponto fixo diferente de zero;
3. Ciclo limite com período 2;
4. Ciclo limite com período 4;
5. Ciclo limite com período 8;
6. Ciclo limite com período 3;
7. Ciclo limite com período 5;
8. Ciclo limite com período 6;
9. Atractor aperiódico (caótico).



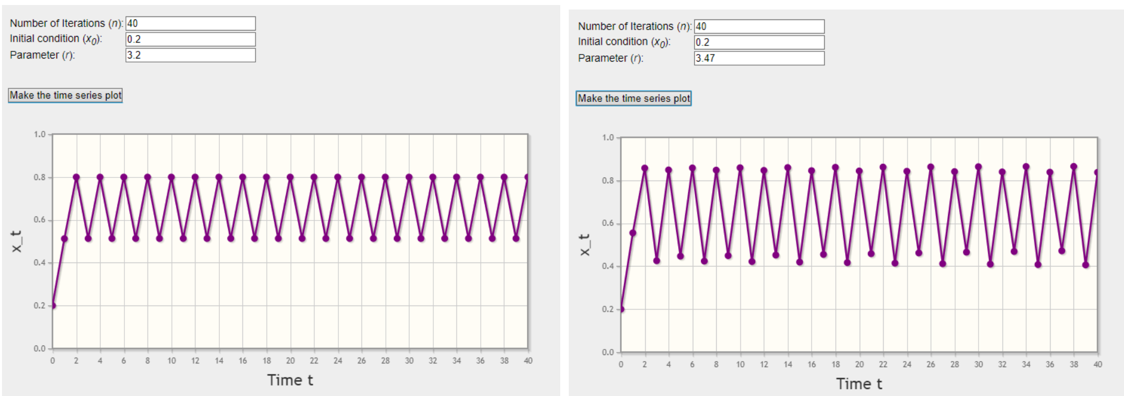
(a) Valor R em que as trajetórias convergem para um ponto fixo 0, R = 0.2

(b) Valor R em que as trajetórias convergem para um ponto fixo 0.5, R = 2

Figura 1: Gráficos alínea 1 e 2

1) Para valores de R entre [0,1] o ponto fixo do sistema é sempre 0. Exemplo para R = 0.5 (Figura 1 a))

2) Para valores de R entre [1,3] o ponto fixo do sistema é sempre um valor exato no eixo vertical. Exemplo para R = 2, Figura 1 a), em que conseguimos observar que o sistema se fixa no ponto 0.5 .



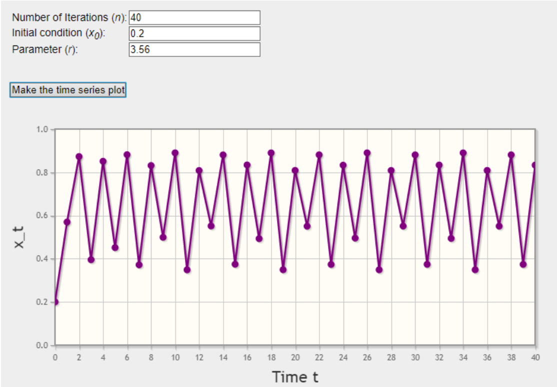
(a) Valor R em que as trajetórias convergem para uma órbita 2, R = 3.2

(b) Valor R em que as trajetórias convergem para uma órbita 4, R = 3.47

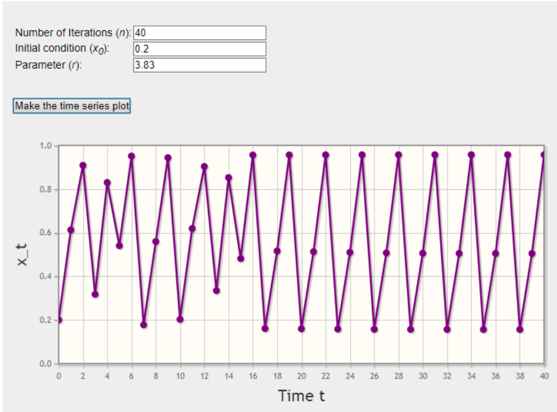
Figura 2: Gráficos alínea 3 e 4

3) Para valores de R entre [3,3.4] o sistema oscila entre 2 valores. Exemplo para R = 3.2 (Figura 2 a)) faz com que a função varie entre 0.5 e 0.8.

4) Para valores de R entre [3.4,3.5] o sistema oscila entre 4 valores. Exemplo para R = 3.47, (Figura 2 b)).



(a) Valor R em que as trajetórias convergem para uma órbita 8, R = 3.56



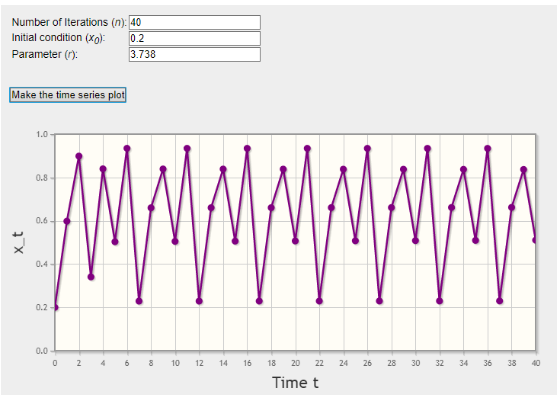
(b) Valor R em que as trajetórias convergem para uma órbita 3, R = 3.83

Figura 3: Gráficos alínea 5 e 6

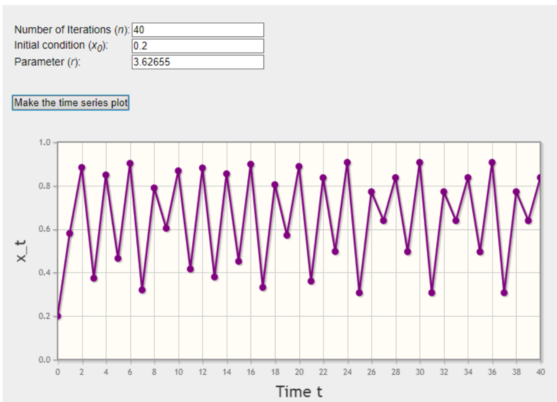
5) Para valores de R entre [3.5,3.6] o sistema oscila entre 8 valores. Exemplo para R = 3.56 (Figura 3 a)).

A partir de r = 3.6 começam a existir demasiadas bifurcações o que torna praticamente impossível a obtenção de qualquer valor,ou seja, estamos na presença de caos. Apesar disto existem momentos em que podemos observar com alguma clareza.

6) Para valores de R entre [3.82,3.84] o sistema vai de completo caos para momentos organizados, oscilando entre 3 valores. Exemplo para R = 3.83, (Figura 3 b)).



(a) Valor R em que as trajetórias convergem para uma órbita 5, R = 3.738



(b) Valor R em que as trajetórias convergem para um ponto fixo 6, R = 3.62655

Figura 4: Gráficos alínea 7 e 8

7) Como anteriormente temos por exemplo R = 3.738 com o qual temos oscilações entre 5 valores (Figura 4 a)).

8) E para R = 3.6255 temos oscilações entre 6 valores (Figura 4 b)).

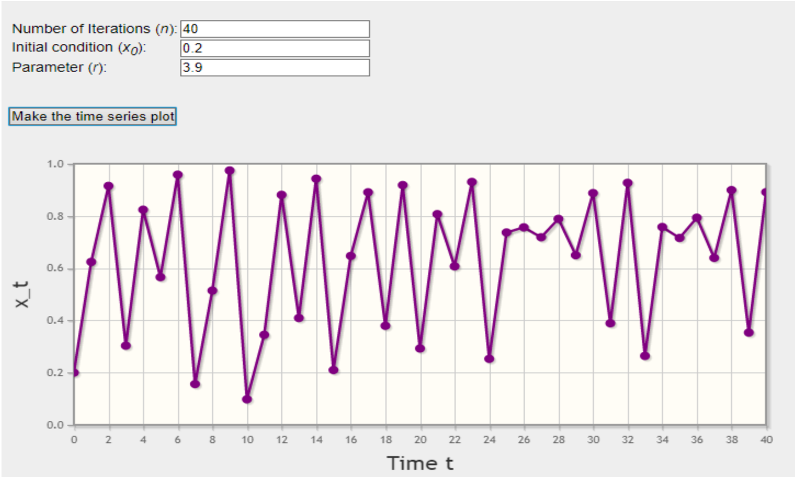


Figura 5: Valor R em que as trajetórias convergem para o caos, R = 3.9

9) Como visto anteriormente a partir de R = 3.6 o caos instala-se, não havendo ciclos periódicos tornando possível a função tomar qualquer valor para R, tirando alguma exceções como as em cima. A partir de 3.86 já não se encontra qualquer oscilação possível. Exemplo para R = 3.9 (Figura 5)

3 B. Jogo do Caos

O Jogo do Caos, na sua forma mais simples, consiste em que num determinado triângulo seja escolhido um ponto arbitrário 'X' no interior desse triângulo. De seguida é escolhido um dos vértices do triângulo 'T' e é pintado no ponto médio entre 'X' e 'T' um ponto da cor do vértice do triângulo escolhido. Este processo é repetido milhares de vezes.

```
public boolean isInsidePolygon(PVector pos)
{
    int i, j=vertices.size()-1;
    int sides = vertices.size();
    boolean oddNodes = false;
    for (i=0; i<sides; i++) {
        if ((vertices.get(i).getCoord().y < pos.y && vertices.get(j).getCoord().y >= pos.y ||
                vertices.get(j).getCoord().y < pos.y && vertices.get(i).getCoord().y >= pos.y) &&
                (vertices.get(i).getCoord().x <= pos.x || vertices.get(j).getCoord().x <= pos.x)) {

            oddNodes^=(vertices.get(i).getCoord().x + (pos.y- vertices.get(i).getCoord().y)/
                    (vertices.get(j).getCoord().y - vertices.get(i).getCoord().y)*
                    (vertices.get(j).getCoord().x- vertices.get(i).getCoord().x)<pos.x);
        }
        j=i;
    }
    return oddNodes;
}
```

Figura 6: Método isInsidePolygon

```
public void JogoDoCaos(PApplet p)
{
    int rando = (int) p.random(vertices.size());
    PVector T = new PVector(vertices.get(rando).getCoord().x, vertices.get(rando).getCoord().y);
    PVector X = new PVector(p.random((float) window[0], (float) window[1]),
            p.random((float) window[2], (float) window[3]));
    while (!isInsidePolygon(X))
    {
        X = new PVector(p.random((float) window[0], (float) window[1]),
                p.random((float) window[2], (float) window[3]));
    }
    PVector y = PVector.mult((X.sub(T)), n: 0.5f);
    T = (T.add(y));

    pontos.add(new Vertice(T, vertices.get(rando).getColor()));

}
```

Figura 7: Método JogoDoCaos

Para a criação do ponto 'X' foi necessário criar uma função que verifica se o ponto criado aleatoriamente está ou não dentro do polígono (figura 1). Depois desse ponto e de um dos vértices do polígono terem sido escolhidos calculamos o ponto médio a partir da função seguinte: X = X + 0.5(T - X) (figura 2). Também foi pedido para em vez de um triângulo ser a forma inicial, o utilizador possa escolher um número arbitrário de pontos que definem os vértices do polígono. Isto foi de fácil execução pois o método da figura 1 permite ser utilizado qualquer polígono, por isso foi apenas necessário fazer com que seja possível o utilizador criar vértices.

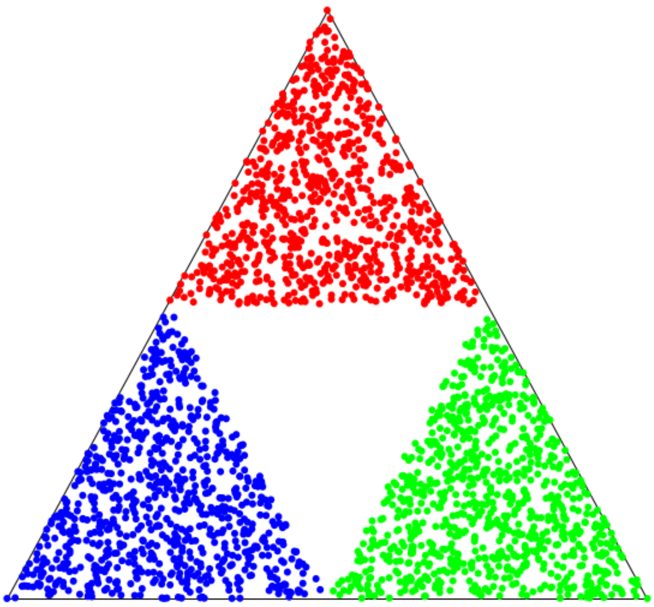


Figura 8: Jogo do Caos num triângulo

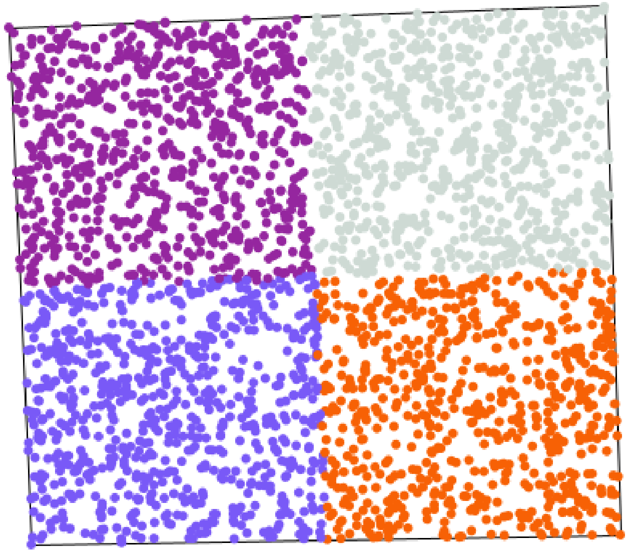
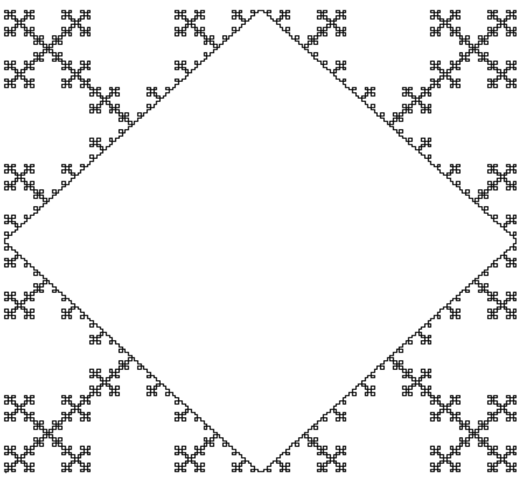


Figura 9: Jogo do Caos num quadrado

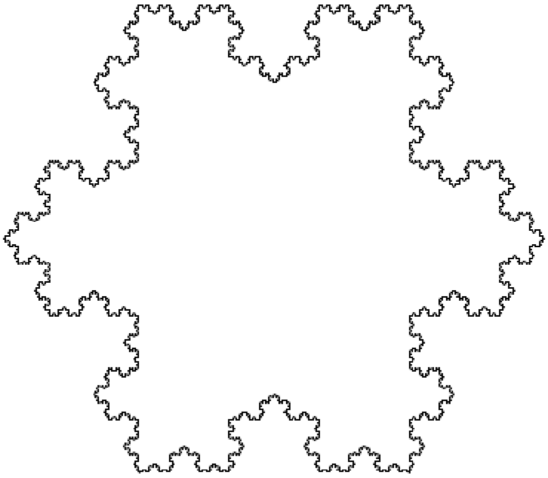
Como podemos observar nas figuras acima (3 e 4), o jogo do caos recria o polígono, em dimensões mais pequenas, tantas vezes quanto o número de vértices da figura principal. Essas cópias vão ter a cor do vértice principal mais próximo e as suas dimensões serão o valor da mediatriz do ponto escolhido a cada um dos pontos.

4 C. Gramáticas de Lindenmayer

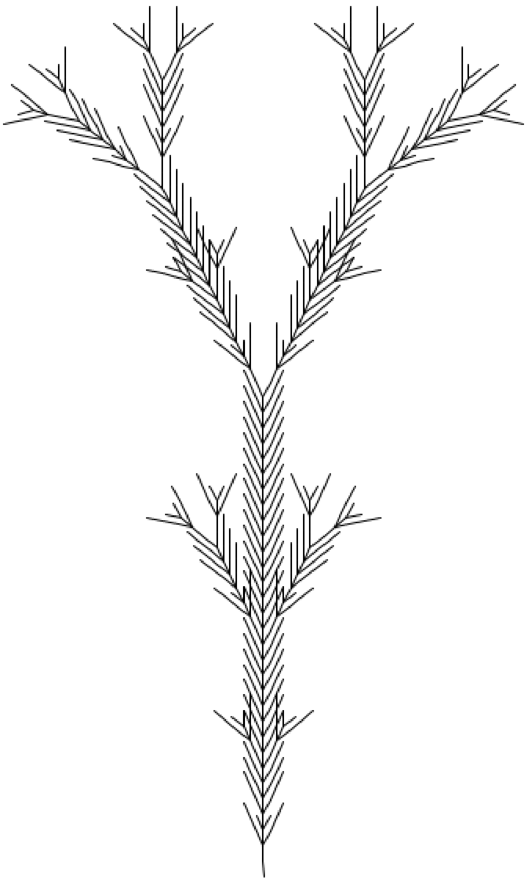
As gramáticas de Lindenmayer ou ”L-Systems”consiste num alfabeto de símbolos que podem ser usados para criar uma cadeia de caracteres, uma coleção de regras de produção que se expandem a cada símbolo numa maior cadeia de símbolos, uma sequência inicial ”axioma”a partir da qual começa a construção, e um mecanismo para traduzir as sequencias geradas em estruturas geométricas. Tendo por base o código feito em aula foi pedido para criar mais dois objetos fractais usando a técnica descrita em cima.



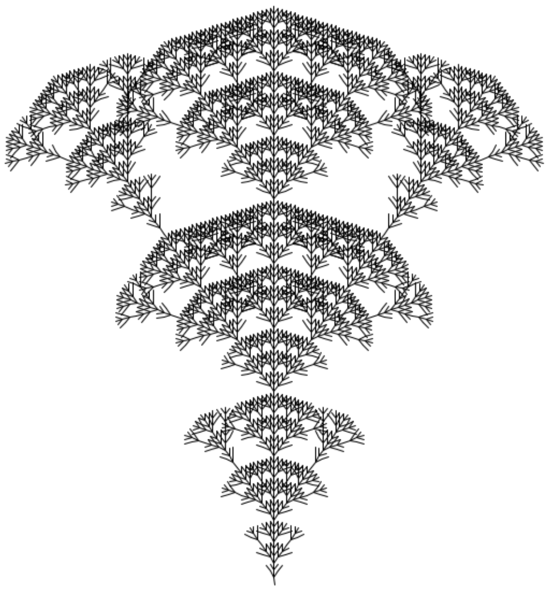
(a) 1º Fractal



(b) 2º Fractal



(c) 3º Fractal



(d) 4º Fractal

Figura 10: Os quatro fractais criados

O primeiro fractal tem o axioma = FF+FF+FF+FF, a regra F -> F+F-F-F+F e o ângulo = 90º. O segundo fractal tem o axioma = F++F++F, a regra F -> F-F++F-F e o ângulo = 60º. O terceiro fractal tem o axioma = Y, as regras X ->X[-FFF][+FFF]FX / Y -> YFX[+Y][-Y] e o ângulo = 25.7º. O quarto fractal tem o axioma = F, a regra F -> F[+FF][-FF]F[-F][+F]F e o ângulo = 35º.

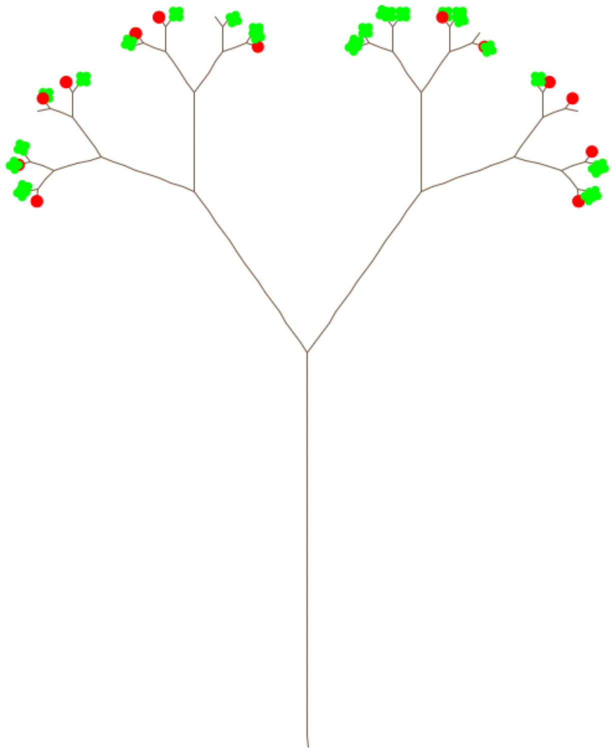


Figura 11: Árvore de frutos

A segunda parte desta pergunta pedia para criar uma árvore de frutos com base no seguinte sistema de Lindemayer: Variáveis: F, G / Constantes: [,], +, - / Axioma: F / Regras: F -> G[+F]-F G -> GG. Sendo necessário então alterar o código de forma a associar ”F”a um ramo terminal, ou seja, existe uma probabilidade de ter folha ou fruto. Também existe a possibilidade de ocorrerem os dois ao mesmo tempo, ou seja, existir folha e fruto no final do mesmo ramo.

5 D. Conjuntos de Julia e Mandelbrot

Nesta fase do trabalho foi necessário, tendo por base o código desenvolvido na aula, experimentar formas alternativas de colorir os pontos que não pertencem ao conjunto de Mandelbrot. A segunda alínea pede para que a janela seja subdividida em duas áreas, sendo que numa é visualizado o conjunto de Mandelbrot e na outra um dado conjunto de Julia. Este conjunto de Julia pode ser escolhido de duas formas diferentes ou pressionamos a tecla ’espaço’ num certo ponto do conjunto de Mandelbrot ou podemos arrastar na janela do conjunto de Julia.

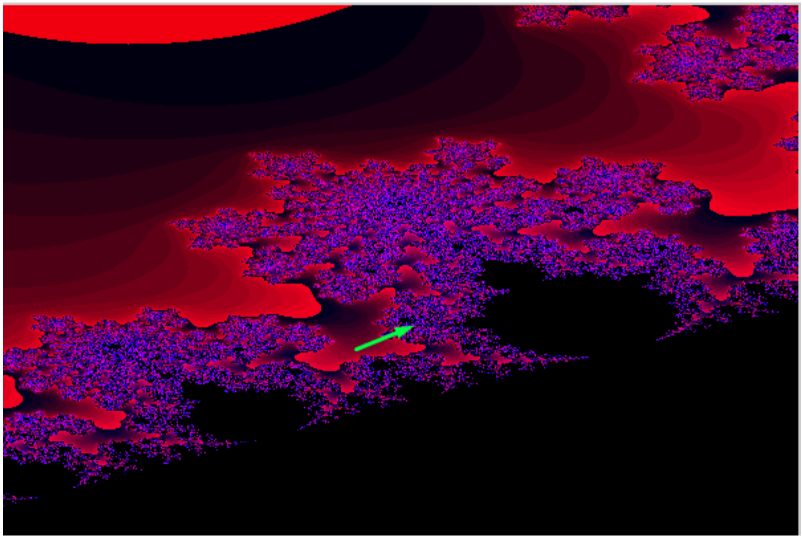


Figura 12: Conjunto de Mandelbrot

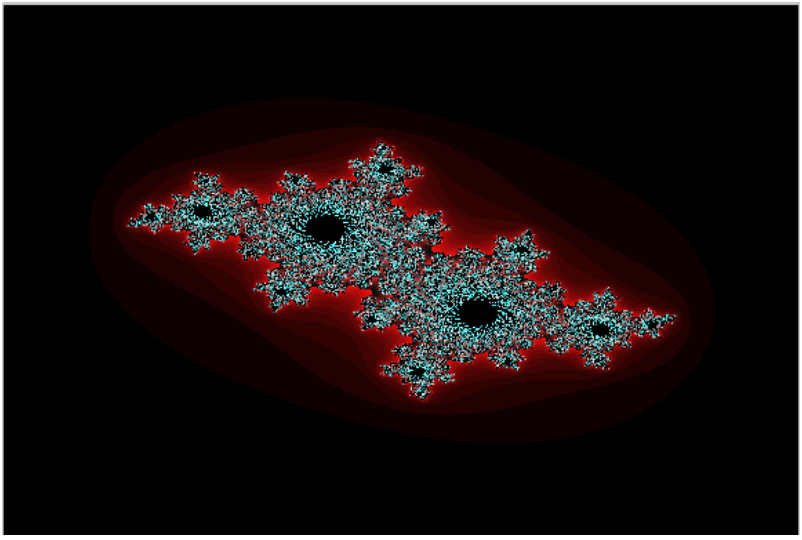


Figura 13: Conjunto de Julia escolhido no conjunto de Mandelbrot

Se formos pela primeira opção ao selecionarmos o ponto no conjunto de Mandelbrot ampliado (seta verde na Figura 7), podemos observar o correspondente conjunto de Julia. Ao percorrermos as extremidades do conjunto de Mandelbrot conseguimos observar que enquanto selecionar-mos os pontos mais próximos do centro apenas vemos preto mas à medida que nos afastamos em determinados conjuntos podemos observar formas mais interessantes como a da Figura 8.

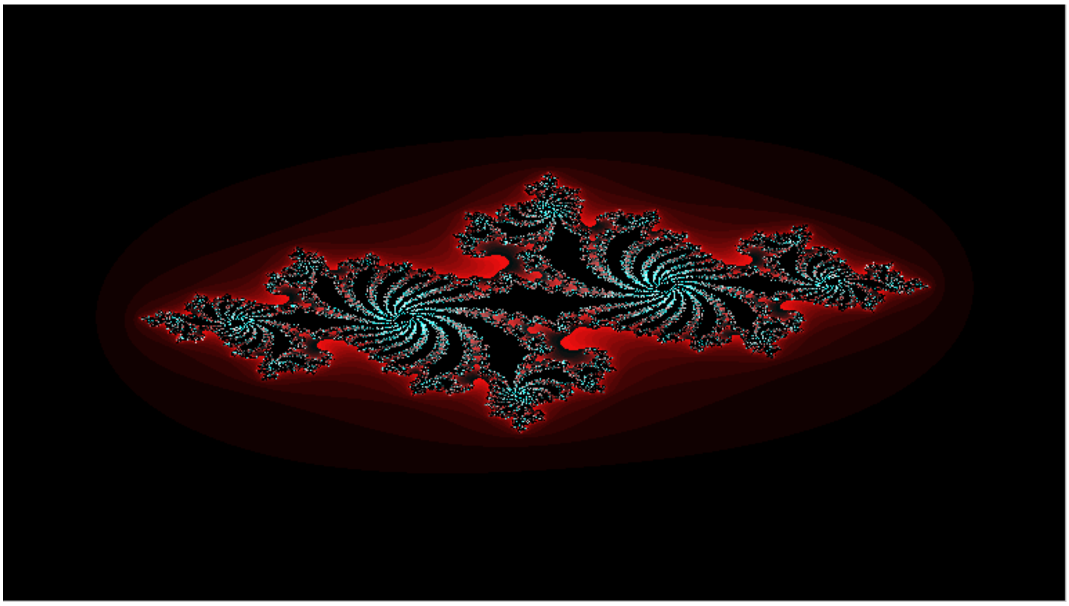


Figura 14: Conjunto de Julia escolhido através do arrasto do rato

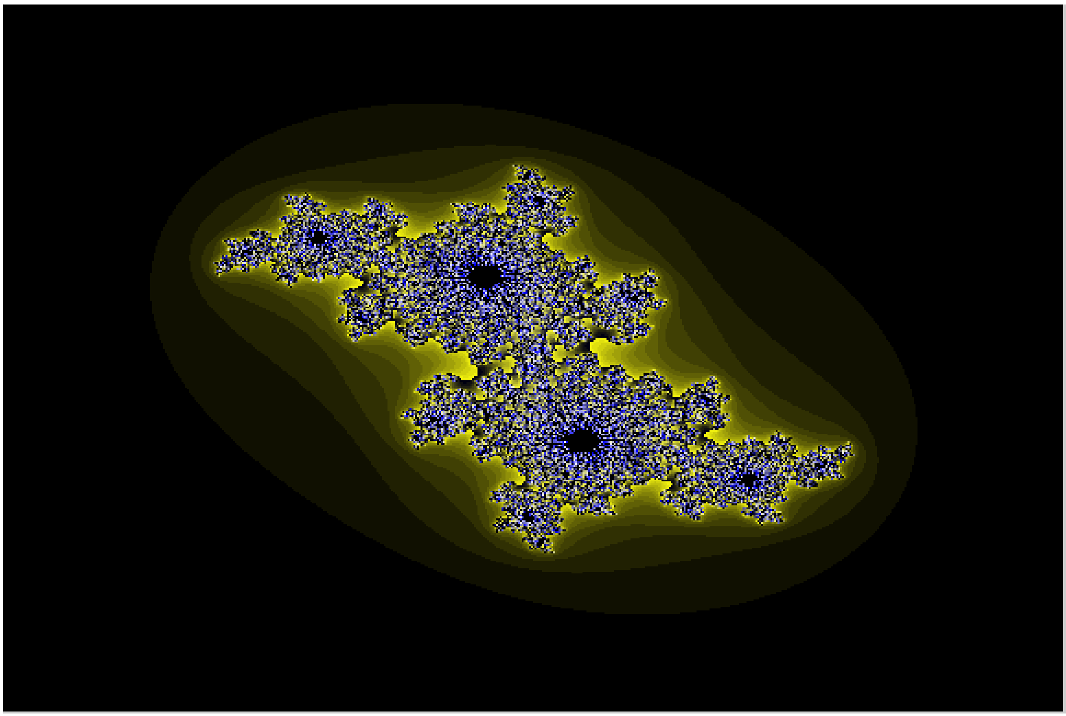


Figura 15: Conjunto de Julia formado por função trigonométrica

Se formos pela segunda opção, sendo menos precisa que a primeira, podemos percorrer livremente o conjunto de Julia e observar as várias formas e cores que são exibidas. Se movimentar-mos o rato ao longo das pontas da janela não existe muito a ser observado mas à medida que nos vamos aproximando do meio e o vamos circulando podemos observar os diferentes padrões possíveis. Também foi adicionado à ferramenta a possibilidade de percorrer as diversas formas possíveis do conjunto de Julia automaticamente, ao substituirmos a parte real da constante 'C' por cos('angle' *3.213) e a parte real sen(angle), em que o 'angle' é uma variável que é incrementada por 0.015 cada vez que o conjunto for desenhado e 3.213 é um número arbitrário podendo ser outro qualquer, mas foi escolhido por dar um aspeto interessante.

6 Conclusões

O objetivo deste projeto era o de estudar uma função logística, criar um Jogo do Caos e consolidar e aprofundar os conhecimentos sobre Gramáticas de Lindenmayer e Conjuntos de Julia e Mandelbrot adquiridos ao longo de Modelação e Simulação de Sistemas Naturais. Gostava de ter tido tempo para desenvolver a pergunta G (Floresta Animada) pois pareceu me uma excelente forma de entrelaçar todos os conhecimentos adquiridos ao longo do semestre nesta disciplina. Porém creio que as aplicações desenvolvidas e as suas partes facultativas estão de acordo com o pretendido pelos docentes, demonstrando o domínio que tenho sobre a matéria lecionada.