

**UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
DEPARTAMENTO DE INFORMÁTICA
ESTRUTURA DE DADOS I**

**GERAÇÃO DE OBJETOS GRÁFICOS ATRAVÉS DE
SISTEMAS DE LINDENMAYER**

**FERNANDO BARBOSA NETO
JEFERSON DE OLIVEIRA BATISTA**

PERÍODO 2015/1

PROF. THOMAS W. RAUBER

**VITÓRIA
28 DE JUNHO DE 2015**

**FERNANDO BARBOSA NETO
JEFERSON DE OLIVEIRA BATISTA**

**GERAÇÃO DE OBJETOS GRÁFICOS ATRAVÉS DE
SISTEMAS DE LINDENMAYER**

Relatório apresentado ao Prof. Thomas W. Rauber
da disciplina de Estrutura de Dados I da
Universidade Federal do Espírito Santo como
requisito parcial para obtenção de média semestral

**VITÓRIA
28 DE JUNHO DE 2015**

SUMÁRIO

RESUMO.....	04
1. INTRODUÇÃO.....	05
2. OBJETIVOS.....	06
3. METODOLOGIA.....	07
4. RESULTADOS DE AVALIAÇÃO.....	08
5. REFERÊNCIAS BIBLIOGRÁFICAS.....	09

RESUMO

Este trabalho visa a criação de um interpretador de um sistema de Lindenmayer, também conhecido como L-System, objetivando a visualização do objeto gráfico gerado.

Para a representação e manipulação da informação estruturada foi utilizada a linguagem C de programação. Foram utilizadas listas, pilhas e árvores para representação das informações necessárias.

Os códigos desenvolvidos estão de acordo com os objetivos traçados pelo trabalho, ou seja, as strings são geradas conforme o esperado para cada parte do trabalho.

1. INTRODUÇÃO

Este trabalho visa a criação um de interpretador de um sistema de Lindenmayer, também conhecido como L-System, objetivando a visualização do objeto gráfico gerado.

Para a representação e manipulação da informação estruturada foi utilizada a linguagem C de programação. Foram utilizadas listas, pilhas e árvores para representação das informações necessárias.

Para o auxílio da implementação e representação gráfica das strings geradas foi utilizada a interface provida pelo Prof. Thomas W. Rauber.

2. OBJETIVOS

- Representação, organização e manipulação de informação estruturada por linguagem de programação de alto nível.
- Desenvolver o conhecimento das estruturas de dados, com o uso de árvores, listas encadeadas, pilhas e dos TAD's (tipos abstratos de dados).
- Com as ferramentas citadas anteriormente, criar um software que seja capaz de gerar o resultado (através de manipulação de strings) de um sistema de Lindenmayer.
- A visualização de objetos gráficos gerados por sistemas de Lindenmayer, através do uso da interface fornecida pelo professor Thomas W. Rauber.

3. METODOLOGIA

O software *lsystem* foi criado utilizando-se a linguagem de alto nível C, com o uso de tipos abstratos de dados (TAD's). Com a biblioteca *arvore.h* foi implementada a árvore com número variável de filhos responsável pelo armazenamento e manipulação da string gerada pela gramática de um sistema de Lindenmayer presente no arquivo de entrada. Na biblioteca *regra.h* está presente a implementação de uma lista encadeada que armazena as regras da gramática do sistema. Temos também a biblioteca *pilha.h* onde está presente a implementação de uma pilha que armazena estados anteriores da geração do objeto gráfico. A biblioteca *lsystem.h* foi responsável por algumas definições usadas no programa. E, finalmente, a biblioteca *psinterface.h* contém a implementação da interface fornecida pelo professor.

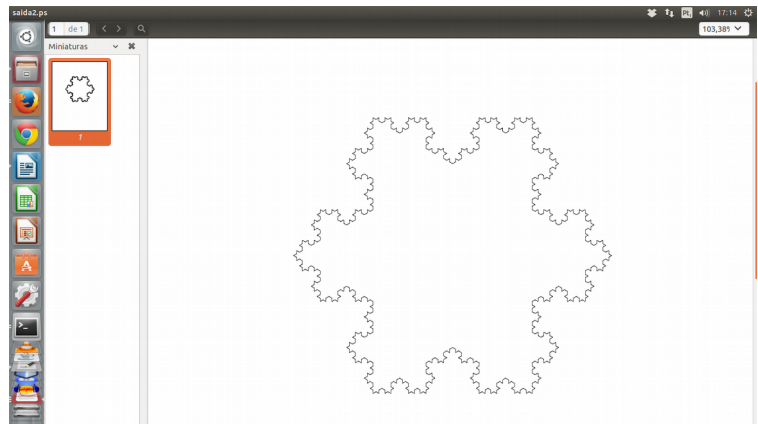
Além das bibliotecas supracitadas, foram usadas as bibliotecas *stdlib.h*, para o uso da alocação dinâmica de memória; a biblioteca *string.h*, para a manipulação de strings lidas no arquivo de entrada e para a manipulação de strings de acordo com o sistema de Lindenmayer lido; a biblioteca *stdio.h*, para a leitura do arquivo de entrada e escrita no arquivo de saída; e a biblioteca *math.h*, para a utilização de algumas operações matemáticas.

4. RESULTADOS E AVALIAÇÃO

Foram utilizados diferentes arquivos de entrada para a realização de testes no software *lssystem*, com o sucesso de geração de diferentes objetos gráficos representando as strings finais resultantes dos sistemas de Lindenmayer presentes nesses arquivos.

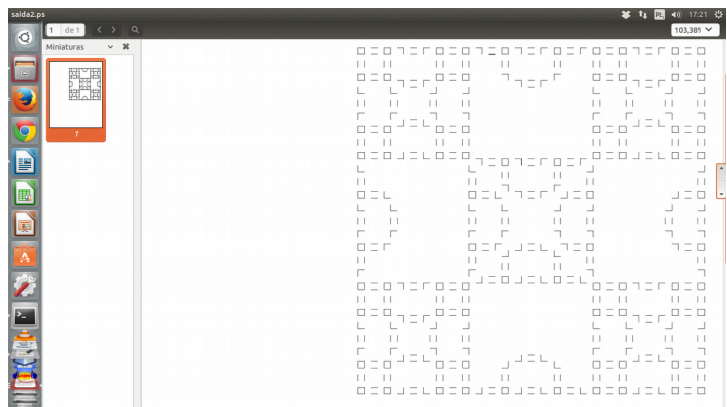
Abaixo encontram-se exemplos de arquivos de entrada e seus respectivos objetos gráficos gerados na saída do programa.

```
; Floco de Neve de Koch  
; preamble  
angle 6 ; means 360/6 degrees  
order 4  
rotate 90  
  
axiom F--F--F  
F = F+F--F+F
```



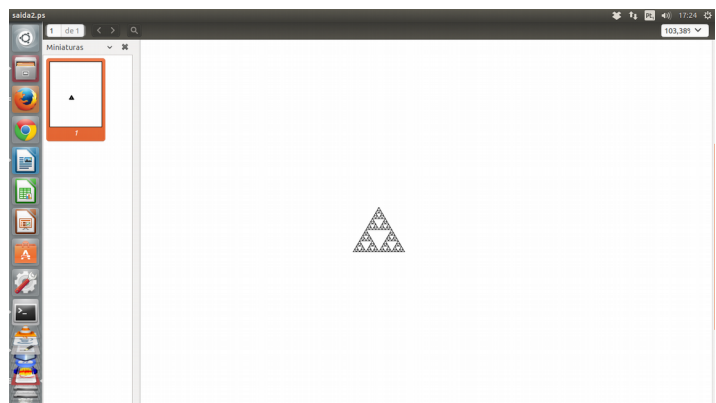
```
; Não tão legal assim...
```

```
angle 4  
order 3  
rotate 0  
  
axiom F+F+F+F  
F = F[GF]+F-GFG-F+F
```



```
; Na real é o Triângulo de Sierpinski
```

```
angle 3  
order 6  
rotate 0  
  
axiom F+F+F  
F = F[+F]F
```



5. REFERÊNCIAS BIBLIOGRÁFICAS

Aulas do professor de Estrutura de Dados I, Thomas W. Rauber, realizadas entre os dias 6 de Maio e 10 de Junho de 2015.

W. Celes, R. Cerqueira, J.L. Rangel. Introdução a Estrutura de Dados, Editora Campus Elsevier, 2004.