

## Disciplina: Introdução à Análise de Algoritmos | P1

### Aluna: Roberta Vitoria Borges nºUSP 11344811 - Turma 04

Este exercício programa tem como objetivo utilizar algoritmos recursivos aplicados à síntese de imagens digitais.

O código do programa foi todo feito na linguagem de programação **Java**.

Para o programa ser executado em seu computador, é necessário ter três arquivos `.java`, sendo eles:

- 1) **Main.java** - Neste código, foi acrescentado os comandos de desenho `KOCH_CURVE`, no qual é especificado 5 valores inteiros para o desenho da Curva de Koch e `REGION_FILL`, no qual é especificado 2 valores inteiros referentes à coordenadas para o preenchimento de uma determinada região.
  - 2) **Image.java** - Não foi modificada
  - 3) **ImageEx.java** - Classe derivada da classe `Image`, nela foi implementada as funcionalidades necessárias para gerar a Curva de Koch e o preenchimento de região (de uma determinada imagem).
- Todos esses arquivos estão disponibilizados dentro da pasta **.zip** entregue.

Para **executar o programa** é necessário seguir os seguintes passos:

- 1) No terminal, acessar a pasta em que se encontra os arquivos
- 2) Rodar `javac Main.java` no terminal
- 3) Rodar `java Main entrada.txt nome_do_arquivo.png`

O arquivo de entrada, como citado no enunciado do Exercício Programa, é um arquivo em formato de texto que contém as definições para a criação de imagens, bem como comandos de desenho a serem executados. O `nome_do_arquivo` pode ser escolhido como preferir, será o nome do arquivo salvo após o programa ser rodado.

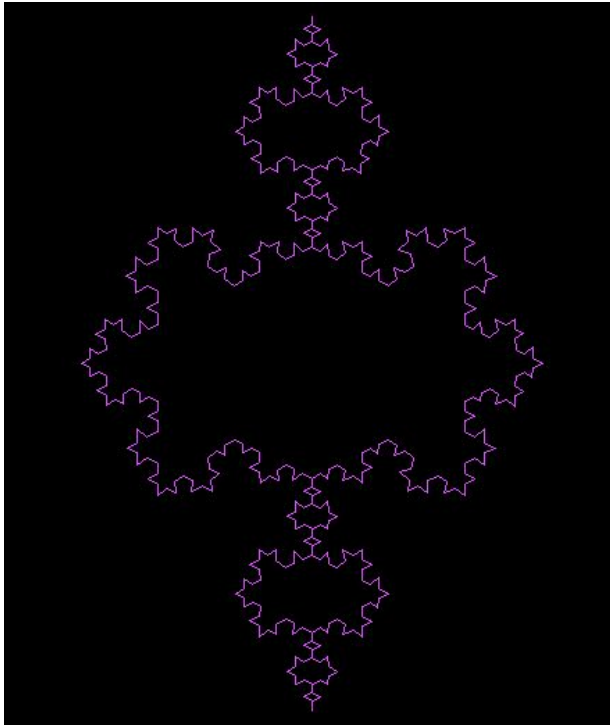
Há um **caso especial** para ser lidado quando executar o comando `REGION_FILL`. Devido o algoritmo recursivo e a quantidade de chamadas feitas para preencher certa região de tamanho `X`, a memória do sistema se esgota facilmente, ocasionando o problema chamado de **StackOverflow**.

Portanto, para executar o comando `REGION_FILL`, é necessário executar  
`java -Xss250M Main entrada.txt nome_do_arquivo.png`

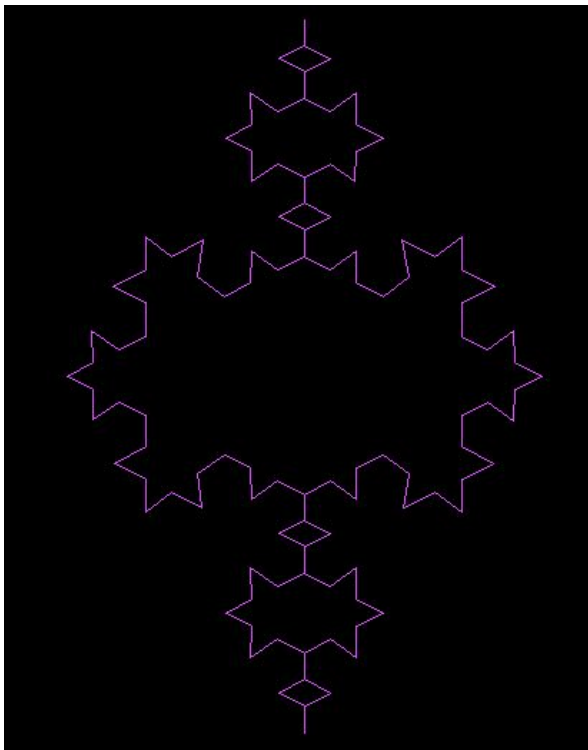
Abaixo estão alguns exemplos que utilizam entradas com o comando `KOCH_CURVE` e `REGION_FILL`. As entradas estão no arquivo **.zip** entregue.

**Saídas da *kochCurve*:**

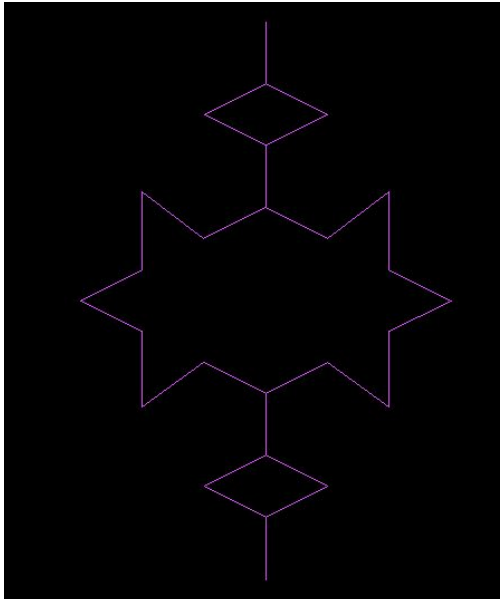
**Entrada 1** - Limiar igual a 5



**Entrada 2** - Limiar igual a 10

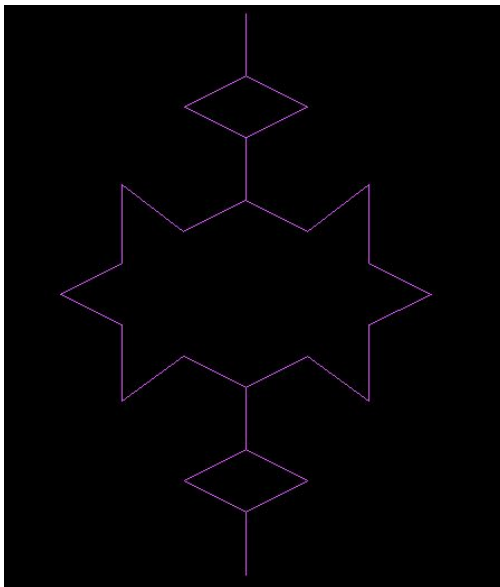


**Entrada 3** - Limiar igual a 20

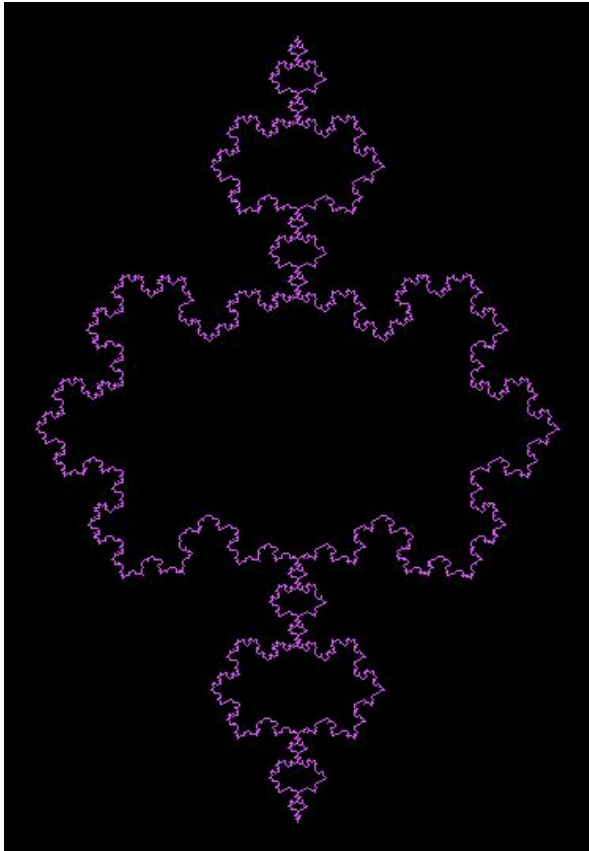


**Entrada 4** - Limiar igual a 25.

Como é possível perceber, a imagem gerada é igual à imagem gerada com o limiar igual a 20, então conclui-se que a partir do limiar igual a 25 todas as imagens geradas são iguais.

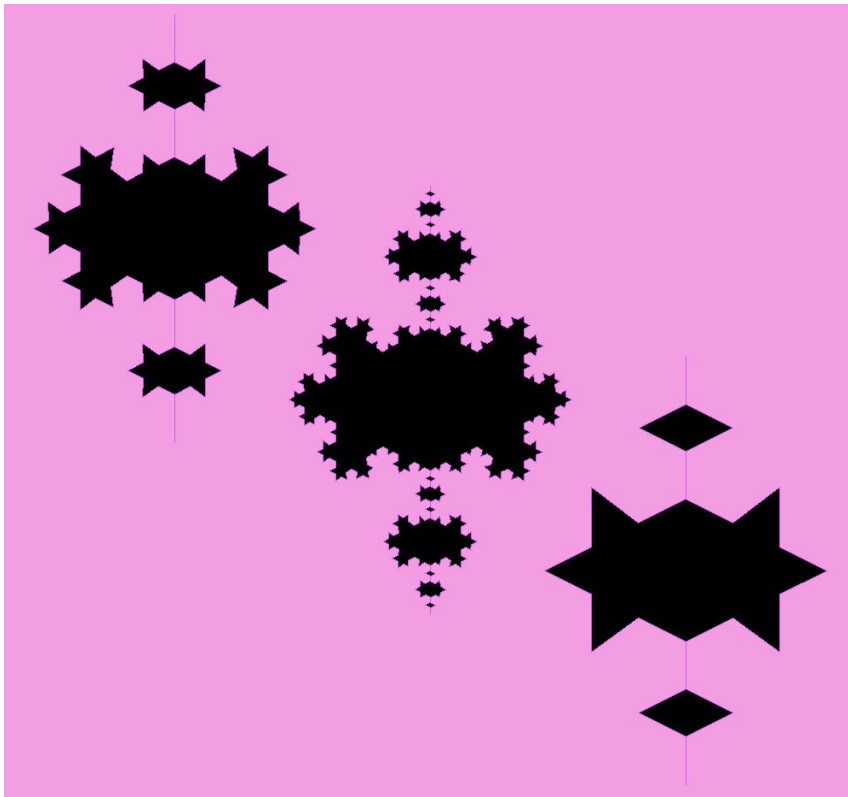


**Entrada 7** - Limiar igual a 1. É o mínimo que pode chegar, quanto menor o limiar, maior a quantidade de triângulos gerados na Curva de Koch, pela limitação dos pixels, a imagem fica difícil de analisar detalhadamente cada curva gerada.

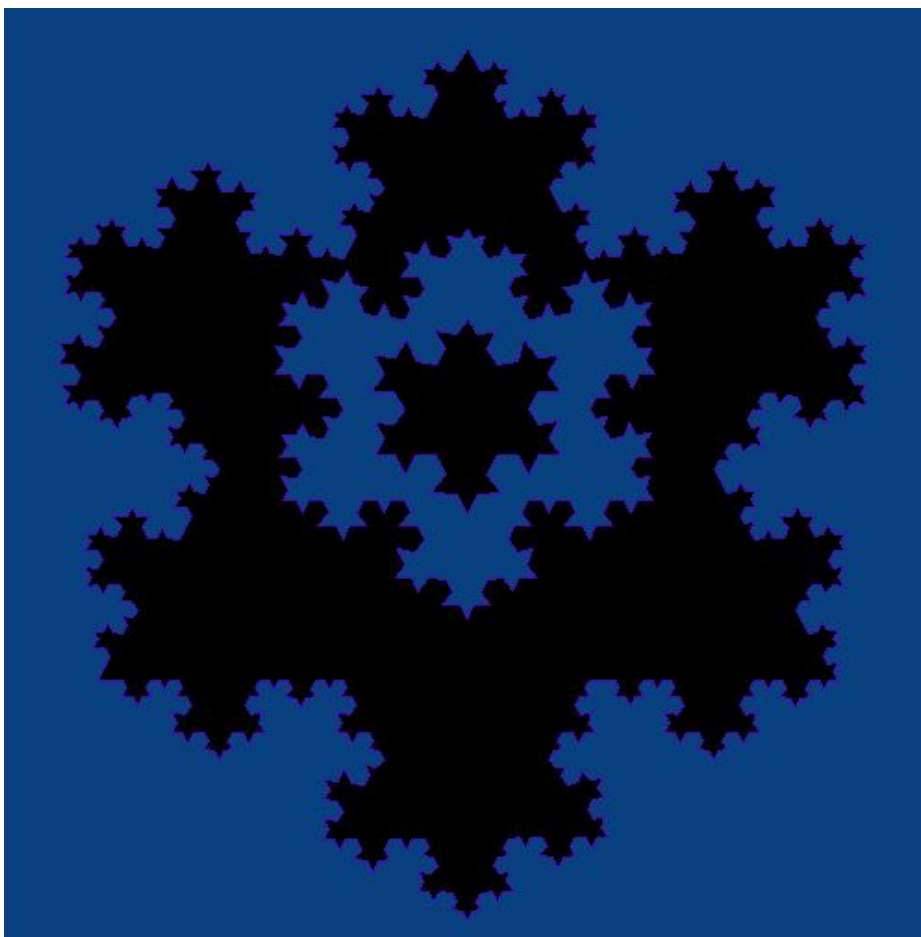


**Saídas da *regionFill* (todas elas utilizando a *kochCurve*):**

**Entrada 5** - Imagem com 3 curvas de Koch, a primeira, da direita para a esquerda com limiar igual a 5, a segunda com limiar igual a 20 e a terceira com limiar igual a 30.



**Entrada 6** - Entrada disponibilizada pelo colega de classe Egydio Pacheco.



**Caso especial:**

Devido aos cálculos e como as retas são formadas, há um caso especial no momento do preenchimento das imagens, como mostrado na imagem abaixo. Duas linhas formam um pixel especial que não consegue ser acessado.

