

## Laboratório 5

Utilizar o template "**template-TurmaC.rkt**" disponível no Moodle.

**Janelas com Desenhos** Para criar desenhos na tela, existem algumas funções pré-definidas, usando o pacote de ensino (*teachpack*) *draw.ss*:

**start** : Number, Number → Boolean

Dados os números de pontos no eixo horizontal e vertical, respectivamente, desenha uma janela com estas dimensões e devolve `true`. O ponto (0,0) desta janela é o canto superior esquerdo. Os argumentos devem ser números inteiros positivos (caso contrário, a chamada incorrerá em um erro de execução).

**draw-circle** : Posn, Number, Symbol → Boolean

Dados o ponto central, o raio (um número inteiro positivo) e a cor, desenha o círculo com estas características na última janela aberta (pela função `start`) e devolve `true`. As cores válidas são estão listadas no arquivo `drcolors.html` no Moodle da disciplina. Caso seja utilizado um nome de cor não válido ou um número não inteiro positivo, ocorrerá um erro de execução.

**draw-solid-disk** : Posn, Number, Symbol → Boolean

Dados o ponto central, o raio (um número inteiro positivo) e a cor, desenha um círculo preenchido (disco) na última janela aberta (pela função `start`) e devolve `true`. Caso seja utilizado um nome de cor não válido ou um número não inteiro positivo, ocorrerá um erro de execução.

**draw-solid-line** : Posn, Posn, Symbol → Boolean

Dados o ponto inicial e o final, respectivamente, e uma cor, desenha uma linha sólida entre os pontos na cor definida e devolve `true`. Caso seja utilizado um nome de cor não válido, ocorrerá um erro de execução.

**draw-solid-rect** : Posn, Number, Number, Symbol → Boolean

Dado um ponto (o canto superior esquerdo de um retângulo), a largura, a altura e a cor de um retângulo, desenha esta figura na última janela aberta e devolve `true`. A largura e a altura devem ser números positivos. Caso os argumentos não sejam números positivos ou uma cor válida, ocorre um erro de execução.

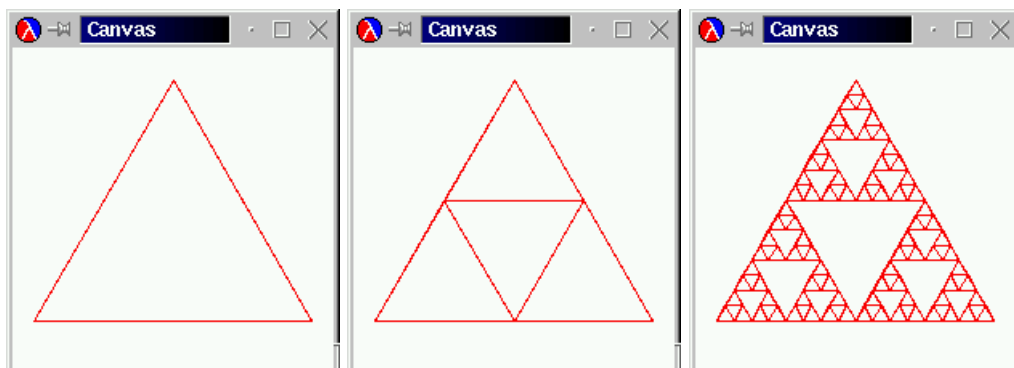
1. Desenvolva a função `draw-triangle`. A função recebe três pontos e uma cor (tipo símbolo) e desenha um triângulo nesta cor na tela. Use `(start 400 400)` para criar a janela antes de testar a função.

```
;; draw-triangle : Posn Posn Posn Symbol -> Boolean
;; ...
```

2. Construa uma função `too-small?` que, dados 3 pontos, define se este triângulo é pequeno demais para ser desenhado na tela. Dica: Usar a função `distance` do template.

```
;; too-small? : Posn Posn Posn -> Boolean
;; ...
```

3. As figuras abaixo mostram 3 etapas da construção de um fractal conhecido como *triângulo de Sierpinski*.



Para gerar esta figura, a idéia é ir dividindo um triângulo em 3 triângulos, e então cada um destes 3 é novamente dividido, e assim sucessivamente. A figura do meio mostra como seria o primeiro passo da divisão. A função que gera um triângulo de Sierpinski deve, a partir de um triângulo inicial, ir subdividindo-o recursivamente até que algum critério de fim seja atingido. Um possível critério seria que o tamanho dos triângulos obtidos fossem pequenos demais para serem subdivididos. Construa o programa `sierpinski`, usando as funções definidas por você no exercícios anteriores (o template contém outras funções que podem também ser usadas).

```
;; sierpinski : Posn Posn Posn Symbol -> Boolean
;; Obj: Desenha um triângulo de Sierpinski com vértices nos pontos passados
;; como argumentos, na cor passada, e devolve true.
;; Se triângulo passado como argumento tiver dimensões muito pequenas,
;; nada é desenhado.
```

Teste o programa com o triângulo com vértices nos pontos A, B, C, D, E e F (ver template).

4. Sobre o programa `sierpinski` desenvolvido no exercício anterior responda as seguintes perguntas:
  - a) Qual a instância trivial do problema?
  - b) Qual a solução para a instância trivial?
  - c) Como os subproblemas mais simples que o original são gerados? Quantos são gerados?
  - d) Como as soluções dos subproblemas são combinadas para gerar a solução do problema original?
  - e) Este programa sempre termina? Por quê?
5. Faça um programa que, dada uma lista de pontos, desenha um triângulo de Sierpinski para cada 3 pontos consecutivos da lista. Se sobraem pontos, estes devem ser ignorados.