

= Roteiro 02 =
Orientação a objetos

Prof. José Rui

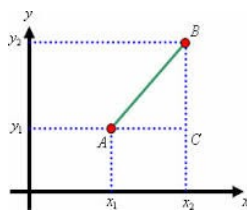
jose.castro@ifsudestemg.edu.br

Objetivo

- Exercitar a construção de um modelo;
- Exercitar o processo de desenvolvimento de um programa de computador a partir da descrição geral de um problema

Atividade 1

Descrição do problema: Seja os seguintes pontos A e B no plano cartesiano abaixo:



Sabemos que, um ponto, é representado/definido por dois valores, X e Y. Desta forma temos que:

A(x1, y1)

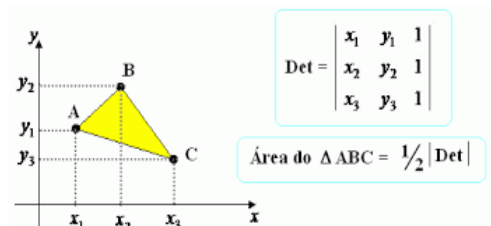
B(x2, y2)

Definimos o seguinte modelo genérico para a classe ponto:

Ponto

-double x
-double y

```
- Ponto();  
- imprimir();  
- distancia(Ponto outro);  
- isColineares(Ponto p2, Ponto p3);  
- areaTriangulo(Ponto p2, Ponto p3);  
- perimetro(Ponto p2, Ponto p3);
```



Vimos que a área do triângulo pode ser encontrada pelo cálculo do determinante conforme imagem. E que, quando, o determinante é zero, significa que os pontos são colineares.

Vamos pensar agora

- Você concorda que os métodos abaixo não fazem sentido estar dentro da classe **Ponto**?

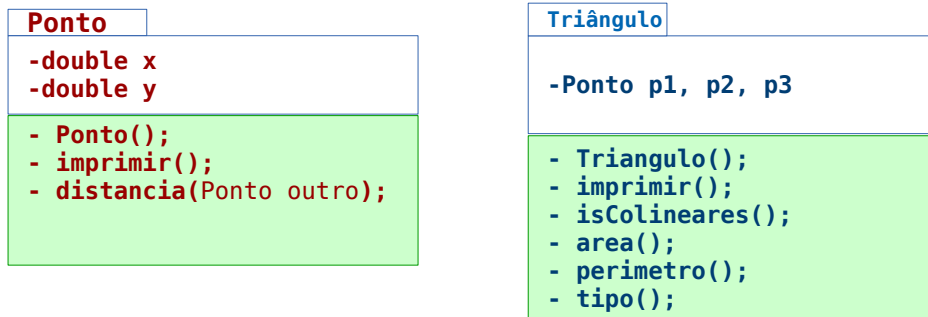
- **isColineares(Ponto p2, Ponto p3);**
- **areaTriangulo(Ponto p2, Ponto p3);**
- **perimetro(Ponto p2, Ponto p3);**

Final de contas, um ponto não tem perímetro, um ponto não tem área e etc. Ou seja, isso não faz

sentido nenhum estar dentro da classe Ponto.

- Na realidade o correto é criar uma classe **Triângulo**, e estes métodos serem colocados nela.

- Então, um possível modelo para o problema seria:



- Agora sim, ficou organizado! Pois os métodos **isColineares(); area(); tipo(); perimetro();** pertencem a classe **Triângulo**.

- O método **tipo();** vai retornar se o **Triângulo** é:

- Equilátero
- Isósceles
- Escaleno

- Perceba que os desenhos dos caixotinhos acima facilitam muito o nosso entendimento. O desenho destes caixotinhos são chamados de modelo. No dia a dia da empresa, sempre criamos o modelo e discutimos o problema olhando para o modelo, ou seja, ele é peça importantíssima no processo. Só depois de criá-lo e discuti-lo que o programador começa a codificar.

Especificação da atividade:

1. Implemente as classes **Ponto** e **Triângulo** conforme o modelo mostrado acima.

2. Execute o programa gerado, verifique se a saída do seu programa está correta. Faça também as correções necessárias, em caso de erros.

Atividade 2

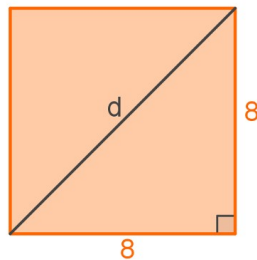
- A partir da classe Ponto crie uma classe Quadrado.
- Sua classe quadrado terá 4 pontos
- O modelo está dado ao lado.

Quadrado
-Ponto p1, p2, p3, p4
<ul style="list-style-type: none">- Quadrado();- imprimir();- area();- perimetro();- tipo();

- Agora pense um pouco, como você saberá que estes 4 pontos formam um quadrado?

Dica:

- Pense no tamanho dos lados
- Pense agora em como descobrir se todos os ângulos formam 90° sem pedir essa informação para o usuário;
- Pense no Triângulo retângulo, na Hipotenusa



Instruções para entrega

Compacte seu pasta **src** com o nome **roteiro02+nomedadupla** e entregue via email/SIGAA.