

Exercício J2_08 de Programação Orientada a Objetos com Java

Sobre a sua solução:

Sua solução **deve conter** ao menos seis arquivos:

- Um arquivo com a função main, com o nome que vocês desejarem, de acordo com as especificações abaixo
- Um arquivo Forma.java
- Um arquivo Circulo.java
- Um arquivo Quadrado.java
- Um arquivo Quadrilatero.java
- Um arquivo Retangulo.java

O script de correção fará dois tipos de testes, onde cada vale parte da "nota":

- Testará o seu código inteiro
- Testará se as suas classes foram implementadas do jeito solicitado (testa ela junto com as outras implementações do professor)

Caso você não implemente todas as classes, ou ainda implemente apenas uma main com saídas pré-definidas, você não receberá nota 10.

Crie a seguinte hierarquia de classes:

- Uma **interface** para representar qualquer **Forma** geométrica, definindo métodos:
 - double calcularPerimetro()
 - double calcularArea()
- Uma **classe abstrata** para representar **Quadriláteros**. Ela deve possuir uma **Array de double** armazenando os 4 lados (não precisa ser inicializada). O método de cálculo do perímetro já pode ser implementado, usando as 4 posições da Array acima;
- Classes para representar Retângulos e Quadrados, que são tipos de Quadriláteros. A primeira deve receber o tamanho da base e da altura (double) no construtor (com métodos getBase e getAltura), enquanto a segunda deve receber apenas o tamanho do lado (com um método getLado); Obs: Se você quiser, pode, mas não é necessário declarar novas variáveis para armazenar essas variáveis.
- Uma classe para representar um Círculo. Seu construtor deve receber o tamanho do raio (número real), e um método getRaio. Use a constante "Math.PI" para calcular raio e perímetro deste círculo.

Escreva, então, um programa que leia dados de formas seguindo as especificações abaixo, armazene as formas em um único vetor e, por fim, imprima, também de acordo com as especificações: (a) os dados (lados ou raio); (b) os perímetros; e (c) as áreas de todas as formas. Para (b) e (c), tire vantagem do polimorfismo, enquanto que para (a) utilize `instanceof` e `downcast`.

Especificações	
Entrada:	Uma linha contendo um número inteiro $N > 0$, seguido de N linhas, cada uma contendo dados de uma forma, no seguinte formato: R,base,altura Q,lado C,raio
Saída:	Dados de cada forma, no formato: descrição da forma – perímetro: ____; area: ____, substituindo as lacunas pelos valores calculados de perímetro e área e descrevendo cada forma com base nos dados de entrada (vide

	exemplos abaixo). Números reais devem ser formatados sempre com 2 casas decimais.
<i>Exemplo de entrada 1:</i>	3 R,10.5,3.75 Q,3.14 C,123.456
<i>Exemplo de saída 1:</i>	Retângulo de base 10.50 e altura 3.75 - perímetro: 28.50; área: 39.38. Quadrado de lado 3.14 - perímetro: 12.56; área: 9.86. Círculo de raio 123.46 - perímetro: 775.70; área: 47882.22.
<i>Exemplo de entrada 2:</i>	10 C,3.875 C,6.338 C,14.496 R,20.699,5.639 Q,16.810 R,12.015,26.136 Q,22.917 R,20.979,18.640 Q,29.693 C,8.730
<i>Exemplo de saída 2:</i>	Círculo de raio 3.88 - perímetro: 24.35; área: 47.17. Círculo de raio 6.34 - perímetro: 39.82; área: 126.20. Círculo de raio 14.50 - perímetro: 91.08; área: 660.16. Retângulo de base 20.70 e altura 5.64 - perímetro: 52.68; área: 116.72. Quadrado de lado 16.81 - perímetro: 67.24; área: 282.58. Retângulo de base 12.02 e altura 26.14 - perímetro: 76.30; área: 314.02. Quadrado de lado 22.92 - perímetro: 91.67; área: 525.19. Retângulo de base 20.98 e altura 18.64 - perímetro: 79.24; área: 391.05. Quadrado de lado 29.69 - perímetro: 118.77; área: 881.67. Círculo de raio 8.73 - perímetro: 54.85; área: 239.43.