
1ª Lista de Exercícios – Atividade Extra-Classe

Data Entrega: vide site

Enviar e-mail:

- Para: poonoite@marciobueno.com
- Assunto: Lista1
- Anexo: Um único arquivo **.ZIP** com todos as pastas dos projetos do Netbeans.

OBS: e-mails fora deste padrão serão desconsiderados

ASSUNTO: Conceitos de orientação a objetos

1. Escreva um modelo para representar uma lâmpada que está à venda em um supermercado. Que dados devem ser representados por este modelo?
2. Crie um modelo Livro que represente os dados básicos de um livro, sem se preocupar com a sua finalidade.
3. Crie um modelo para representar um time de um esporte qualquer em um campeonato desse esporte. Que dados e operações esse modelo deve ter?
4. Crie um modelo Ponto2D para representar um ponto no espaço cartesiano de duas dimensões. Que dados e operações esse modelo deve ter? *Dica:* Imagine um gráfico no qual você tenha que desenhar pontos, baseados nesse modelo.
5. Crie um modelo Veiculo que represente os dados e operações de um veículo de transporte como nome, número de passageiros que pode carregar, tipo (aéreo, terrestre ou aquático), número de rodas etc.
6. Com base no exercício 5, crie um modelo VeiculoTerrestre que tenha os mesmos dados do modelo Veiculo. Quais serão as principais diferenças entre o modelo criado nesse exercício e o modelo Veiculo?
7. Escreva um modelo Empregado que represente um empregado de uma empresa qualquer. Considere que os dados nome, departamento, horasTrabalhadasNoMês e salárioPorHora devam ser representados, e que ao menos as operações mostraDados e calculaSalárioMensal sejam implementadas.
8. Crie um modelo para representar uma linha, unida por dois pontos no espaço cartesiano de duas dimensões, usando o modelo criado no exercício 4. Que dados e operações esse modelo deve ter?
9. Imagine que os empregados de uma empresa tenham dois valores de salário para horas trabalhadas, diferenciados entre horas normais e horas extras. Modifique o modelo Empregado (veja o exercício 7) para que dois valores de horas trabalhadas e dois valores de salário-hora sejam usados.

ASSUNTO: Implementação dos conceitos de orientação a objetos

10. Escreva a classe Ponto2D, correspondente ao modelo da resposta do exercício 4.
11. Escreva a classe Veiculo, correspondente ao modelo da resposta do exercício 5.
12. Escreva a classe VeiculoTerrestre, correspondente ao modelo da resposta do exercício 6. Veja também o exercício 11.
13. Escreva uma classe Livro que represente o modelo do exercício 2.
14. Escreva uma classe AutomovelUsado que represente os dados de um automóvel usado à venda, como ano, modelo, quilometragem rodada, combustível, preço pedido etc. Que campos e métodos essa classe deve ter?
15. Escreva uma classe CadernoDeEnderecos que represente os dados de uma pessoa, como nome, telefone, e-mail e endereço. Que campos e métodos essa classe deve ter?
16. Escreva a classe Contador que encapsule um valor usado para contagem de itens ou eventos. Essa classe deve esconder o valor encapsulado de programadores-usuários, fazendo com que o acesso ao valor seja feito através de métodos que devem zerar, incrementar e imprimir o valor do contador.
17. Escreva a classe Empregado, correspondente à resposta do exercício 7.
18. Crie uma classe Linha para representar uma linha, unida por dois pontos no espaço cartesiano de duas dimensões, usando duas instâncias da classe Ponto2D, criada no exercício 10. Veja também o exercício 8.

ASSUNTO: Aplicações em Java – método main

19. Explique, com suas palavras, por que uma classe como a Ponto2D (exercício 10) não pode ser executada.
20. Escreva um programa em Java que imprima o seu nome.
21. Escreva um programa em Java que leia o seu nome do teclado e imprima-o com uma mensagem qualquer.
22. Escreva uma aplicação em Java que demonstre o uso de instâncias da classe Contador, que deve ter sido criada como resposta ao exercício 16.
23. Usando as classes Veiculo e VeiculoTerrestre, que devem ter sido criadas como resposta aos exercícios 11 e 12, escreva uma aplicação em Java que declare várias instâncias das duas classes. Em particular, tente descrever o mesmo veículo usando duas instâncias, uma da classe Veiculo e outra da classe VeiculoTerrestre.

ASSUNTO: Mais métodos

24. O que acontecerá se tentamos imprimir uma instância de uma classe que não tem o método toString? Demonstre com um pequeno programa.
25. Crie o método `criaRevertido` para a classe `Ponto2D` (exercício 10) que retorne uma nova instância da classe onde os valores encapsulados `x` e `y` são revertidos.
26. Escreva o método `toString` para a classe `Linha` (exercício 18), reaproveitando o método `toString` da classe `Ponto2D` (exercício 10).
27. Crie o método `temEixoComum` para a classe `Ponto2D` (exercício 10) que receba uma outra instância da classe `Ponto2D` e retorne o valor booleano `true` se as coordenadas horizontais e/ou verticais encapsuladas forem iguais às da instância passada como argumento. Por exemplo, as coordenadas (1,2) e (1,-17) têm eixo comum, as (-9,0) e (9,0) também têm, mas as (13,-8) e (8,-22) não têm eixo comum.
28. Crie o método `distância` para a classe `Ponto2D` (exercício 10) que recebe uma outra instância da classe `Ponto2D` e retorna um valor do tipo `double` correspondente à distância euclidiana entre o `Ponto2D` encapsulado e o passado como argumento. Dica: A distância euclidiana d entre um ponto com coordenadas (x_1, y_1) e outro ponto com coordenadas (x_2, y_2) é calculada por $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$, que pode ser escrita em Java como $d = \text{Math.sqrt}((x_1 - x_2) * (x_1 - x_2) + (y_1 - y_2) * (y_1 - y_2))$, onde `Math.sqrt` é o método em Java que calcula a raiz quadrada.
29. Crie o método `distânciaDaOrigem` para a classe `Ponto2D` (exercício 10) que não recebe nenhum argumento, mas calcula a distância euclidiana entre as coordenadas encapsuladas e a origem do sistema de coordenadas. Para isso, dentro do método, crie uma instância de `Ponto2D` correspondente à origem e passe-a como argumento para o método `distância`, que deve ter sido escrito como resposta ao exercício 28.
30. Crie o método `ehProximo` para a classe `Ponto2D` (exercício 10) que recebe uma outra instância da classe `Ponto2D` e um limiar (valor do tipo `double`) como argumentos, calculando a distância entre as coordenadas encapsuladas e as coordenadas da instância passada como argumento, retornando o valor booleano `true` se a distância for menor do que o limiar passado como argumento. Por exemplo, se o ponto encapsulado vale (3,3), o passado como argumento vale (4,1). Se o limiar for 3.0, o método deve retornar `true` já que a distância entre os dois pontos (2.236) é menor do que o limiar. Se o limiar fosse 2.0, o método deverá retornar `false`.

ASSUNTO: Construtores

31. Escreva dois construtores para a classe `Contador`, um que não receba argumentos e considere que o contador começa a contar a partir do zero, e outro que aceita um valor inicial para contagem.
32. Escreva dois construtores para a classe `Ponto2D` (exercício 10): um sem argumentos que considere que o ponto está na origem, ou seja, com coordenadas (0,0), e um que receba dois argumentos do tipo `double` e que os use para inicializar os campos da classe.
33. Escreva quatro construtores para a classe `Linha` (exercício 2.35): um sem argumentos que considere que a linha comece e termine no ponto (0,0); um que receba um argumento do tipo `Ponto2D` e que considere que a linha comece na origem e termine no ponto passado como argumento; um que receba

duas instâncias da classe Ponto2D como argumentos e um que receba quatro valores de ponto flutuante, correspondentes às duas coordenadas.

ASSUNTO: Campos e métodos estáticos

34. Explique, com suas palavras, por que os campos na classe ConstantesMatematicas (slide 11 de POO_09_AtributosMetodos_Estaticos) não devem ser declarados com o modificador private.
35. Escreva a classe ConversaoDeUnidadesDeArea com métodos estáticos para conversão das unidades de área segundo a lista abaixo.
 - 1 metro quadrado = 10.76 pés quadrados
 - 1 pé quadrado = 929 centímetros quadrados
 - 1 milha quadrada = 640 acres
 - 1 acre = 43.560 pés quadrados
36. Escreva a classe ConversaoDeUnidadesDeVolume com métodos estáticos para conversão das unidades de volume segundo a lista abaixo.
 - 1 litro = 1000 centímetros cúbicos
 - 1 metro cúbico = 1000 litros
 - 1 metro cúbico = 35.32 pés cúbicos
 - 1 galão americano = 231 polegadas cúbicas
 - 1 galão americano = 3.785 litros
37. Escreva a classe ConversaoDeUnidadesDeTempo com métodos estáticos para conversão aproximada das unidades de velocidade segundo a lista abaixo.
 - 1 minuto = 60 segundos
 - 1 hora = 60 minutos
 - 1 dia = 24 horas
 - 1 semana = 7 dias
 - 1 mês = 30 dias
 - 1 ano = 365.25 dias
38. Escreva uma classe que contenha métodos estáticos para retornar o maior e o menor de dois, três, quatro e cinco valores (com um total de oito métodos), considerando que os argumentos e retorno dos métodos podem ser dos tipos int e double. Dica: Os métodos podem ser chamados em cascata: para calcular o maior de três valores a, b e c, pode-se calcular o maior valor de a e b, e comparar esse resultado com c.
39. Escreva uma classe que contenha métodos estáticos para calcular as médias e somas de dois, três, quatro e cinco valores, considerando que os argumentos e retorno dos métodos podem ser dos tipos int e double. Um total de 16 métodos deverão ser criados.
40. Escreva uma classe SerieLimitada, que encapsula um valor inteiro seqüencial como os usados em notas e séries de gravuras. Essa classe deve permitir que um programa crie um número limitado de instâncias dela, cada uma numerada com um valor seqüencial. O número total de instâncias é controlado pelo campo máximoDeInstâncias, declarado como static final, e o de instâncias já criadas é controlado pelo campo contador declarado como static. Escreva também uma aplicação que crie algumas instâncias da classe para demonstrar seu funcionamento.