

Universidade de Estado do Amazonas - UEA
Lista de exercícios

Para todos os exercícios abaixo deve-se, além de atender o enunciado do próprio exercício, também escrever a classe pública com o método **main** que irá instanciar pelo menos quatro objetos de cada classe, fazendo uso de todos os atributos e métodos das mesmas. Deve-se ainda elaborar a representação gráfica (UML) cada uma das classes.

- 1) Escreva um código em Java que apresente a classe Pessoa, com atributos nome, endereço e telefone e, o método imprimir. O método imprimir deve mostrar na tela os valores de todos os atributos.
- 2) Baseando-se no exercício 1 adicione um método construtor que permita a definição de todos os atributos no momento da instanciação do objeto.
- 3) Escreva um código em Java que apresente a classe Quadrado, com atributos lado, área e perímetro e, os métodos calcularArea, calcularPerimetro e imprimir. Os métodos calcularArea e calcularPerimetro devem efetuar seus respectivos cálculos e colocar os valores nos atributos área e perímetro. O método imprimir deve mostrar na tela os valores de todos os atributos. Salienta-se que a área de um quadrado é obtida pela fórmula (lado * lado) e o perímetro por (4 * lado).
- 4) Baseando-se no exercício 3 adicione um método construtor que permita a definição de todos os atributos no momento da instanciação do objeto.
- 5) No mesmo projeto da questão 3, crie uma classe que apresente um Retângulo, com atributos comprimento, largura, área e perímetro e, os métodos calcularArea, calcularPerimetro e imprimir. Os métodos calcularArea e calcularPerimetro devem efetuar seus respectivos cálculos e colocar os valores nos atributos area e perimetro. O método imprimir deve mostrar na tela os valores de todos os atributos. Salienta-se que a área de um retângulo é obtida pela fórmula (comprimento * largura) e o perímetro por (2 * comprimento) + (2 * largura).
- 6) Baseando-se no exercício 5 adicione um método construtor que permita a definição de todos os atributos no momento da instanciação do objeto.
- 7) Crie outra classe que represente a forma geométrica Circulo, com atributos raio, área e perímetro e, os métodos calcularArea, calcularPerimetro e imprimir. Os métodos calcularArea e calcularPerimetro devem efetuar seus respectivos cálculos e colocar os valores nos atributos area e perimetro. O método imprimir deve mostrar na tela os valores de todos os atributos. Salienta-se que a área de um círculo é obtida pela fórmula (pi * raio * raio) e o perímetro por (2 * pi * raio), onde pi = 3,141516.
- 8) Baseando-se no exercício 7 adicione um método construtor que permita a definição de todos os atributos no momento da instanciação do objeto.
- 9) Escreva um novo projeto em Java. Neste projeto crie um código que apresente a classe Moto, com atributos marca, modelo, cor e marcha e, o método imprimir. O método imprimir deve mostrar na tela os valores de todos os atributos. O atributo marcha indica em que a marcha a Moto se encontra no momento, sendo representado de forma inteira, onde 0 - neutro, 1 – primeira, 2 – segunda, etc.
- 10) Baseando-se no exercício 9 adicione um método construtor que permita a definição de todos os atributos no momento da instanciação do objeto.

- 11) Baseando-se no exercício 10 adicione os métodos `marchaAcima` e `marchaAbaixo` que deverão efetuar a troca de marchas, onde o método `marchaAcima` deverá subir uma marcha, ou seja, se a moto estiver em primeira marcha, deverá ser trocada para segunda marcha e assim por diante. O método `marchaAbaixo` deverá realizar o oposto, ou seja, descer a marcha. O método `imprimir` deve ser modificado de forma a mostrar na tela os valores de todos os atributos.
- 12) Baseando-se no exercício 11 adicione os atributos `menorMarcha` e `maiorMarcha`, onde o atributo `menorMarcha` indica qual será a menor marcha possível para a moto e o atributo `maiorMarcha` indica qual será a maior marcha possível. Desta forma os métodos `marchaAcima` e `marchaAbaixo` devem ser reescritos de forma a não permitirem a troca de marchas para valores abaixo da `menorMarcha` e acima da `maiorMarcha`. O método `imprimir` deve ser modificado de forma a mostrar na tela os valores de todos os atributos.
- 13) Baseando-se no exercício 12 adicione um método construtor que permita a definição de todos os atributos no momento da instanciação do objeto.
- 14) Baseando-se no exercício 13 adicione o atributo `ligada` que terá a função de indicar se a moto está ligada ou não. Este atributo deverá ser do tipo booleano. O método `imprimir` deve ser modificado de forma a mostrar na tela os valores de todos os atributos.
- 15) Baseando-se no exercício 14 adicione um método construtor que permita a definição de todos os atributos no momento da instanciação do objeto.
- 16) Baseando-se no exercício 15 adicione os métodos `ligar` e `desligar` que deverão mudar o conteúdo do atributo `ligada` conforme o caso.
- 17) Escreva um projeto em Java que defina a classe `Eletrodomestico`, com atributo `ligado` e o método `imprimir`. O método `imprimir` deve mostrar na tela os valores de todos os atributos. O atributo `ligado` será booleano e deverá indicar o estado atual do eletrodoméstico, se ligado ou desligado.
- 18) Baseando-se no exercício 17 adicione um método construtor que permita a definição de todos os atributos no momento da instanciação do objeto.
- 19) Baseando-se no exercício 18 adicione os métodos `ligar` e `desligar` que deverão mudar o conteúdo do atributo `ligado` conforme o caso.
- 20) Escreva um código em Java que apresente a classe `Televisor`, com atributos `ligado`, `canal` e `volume` e, o método `imprimir`. O método `imprimir` deve mostrar na tela os valores de todos os atributos. O atributo `ligado` será booleano e deverá indicar o estado atual do televisor, se ligado ou desligado. O atributo `canal` deverá indicar o canal atual em que o televisor está sintonizado. O atributo `volume` deverá indicar o volume atual do televisor.
- 21) Baseando-se no exercício 20 adicione um método construtor que permita a definição de todos os atributos no momento da instanciação do objeto.
- 22) Baseando-se no exercício 21 adicione os métodos `ligar` e `desligar` que deverão mudar o conteúdo do atributo `ligado` conforme o caso.
- 23) Baseando-se no exercício 22 adicione os atributos `numeroCanais` e `volumeMaximo`, onde `numeroCanais` indica o número máximo de canais que o televisor pode sintonizar

e volumeMaximo indica qual o maior nível de volume possível. O método imprimir deve ser modificado de forma a mostrar na tela os valores de todos os atributos.

- 24) Baseando-se no exercício 23 adicione os métodos canalAcima e canalAbaixo, sendo que o método canalAcima deve sintonizar sempre o próximo canal em relação ao canal atual, onde ao chegar ao maior canal possível deverá voltar ao canal 1. O método canalAbaixo deve sintonizar sempre o canal anterior em relação ao canal atual, onde ao chegar ao canal 1 deverá voltar ao maior canal possível, simulando desta forma o funcionamento de um televisor.
- 25) Baseando-se no exercício 24 adicione os métodos volumeAcima e volumeAbaixo, sendo que o método volumeAcima deve modificar o volume para o próximo nível de volume possível, onde ao chegar ao volumeMaximo não poderá possibilitar que o volume seja alterado. O método volumeAbaixo deve modificar o volume para o nível imediatamente inferior de volume em relação ao volume atual, não podendo ser menor do que 0, simulando desta forma o funcionamento de um televisor.
- 26) Escreva um código em Java que apresente a classe Microondas, com atributo ligado e o método imprimir. O método imprimir deve mostrar na tela os valores de todos os atributos. O atributo ligado será booleano e deverá indicar o estado atual do microondas, se ligado ou desligado.
- 27) Baseando-se no exercício 26 adicione um método construtor que permita a definição de todos os atributos no momento da instanciação do objeto.
- 28) Baseando-se no exercício 27 adicione os métodos ligar e desligar que deverão mudar o conteúdo do atributo ligado conforme o caso.
- 29) Baseando-se no exercício 28 adicione o atributo portaFechada que deverá ser booleano e deverá indicar se a porta do microondas está ou não fechada. O método imprimir deve ser modificado de forma a mostrar na tela os valores de todos os atributos.
- 30) Baseando-se no exercício 29 adicione os métodos fecharPorta e abrirPorta que deverão mudar o conteúdo do atributo portaFechada conforme o caso.
- 31) Baseando-se no exercício 30 modifique o método ligar para que só ligue o equipamento quando a porta do mesmo estiver fechada, simulando assim o funcionamento de um microondas.
