

Universidade Federal do Agreste de Pernambuco

Curso: Bacharelado em Ciência da Computação.

Disciplina: Programação Orientada à Objetos.

Professora: Thaís Alves Burity Rocha.

2ª Lista de Exercícios

**Sempre que possível,
primeiro projete o diagrama
de classes da UML**

Exercícios práticos

- 1) (Rafael Santos) Usando UML, projete uma classe 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.
- 2) (Ana Cristina Melo) Gabriel usa a planilha a seguir para controlar os gastos mensais com sua conta de luz. Mensalmente Gabriel verifica os meses de maior e menor consumo. Usando UML, identifique a classe desse cenário, com seus atributos e métodos.

Data leitura	Nº leitura	Kw gasto	Valor a pagar	Data pgto	Média consumo
04/07/2023	4166	460	206,43	15/07/2023	15,33
02/08/2023	4201	350	157,07	15/08/2023	12,06

- 3) (Ana Cristina Melo, adaptada) Usando UML, projete a classe que permite configurar um texto por meio de atributos de tamanho da letra, cor da fonte, cor do fundo e o tipo de componente em que ele deve ser exibido.
- 4) (Ana Cristina Melo, adaptada) As informações a seguir se referem à uma aplicação de controle pessoal de horário de remédios. Nessa aplicação, para cada remédio, cadastrase o nome de quem vai tomar o remédio, a data de início, a quantidade de dias que foi prescrita pelo médico, a quantidade de vezes ao dia, a dosagem e o nome do remédio. A aplicação sugere para o usuário os horários para tomar o remédio, considerando a prescrição e os horários de acordar e dormir. Além disso, também gera uma planilha de horários de remédios do dia. Usando UML, identifique a classe, atributos e métodos dessa aplicação. Suponha a existência das classes Date para data e Time para horário.
- 5) (Ana Cristina Melo) Vera usa uma planilha para fazer o controle dos seus gastos diários. Para cada gasto, Vera cadastra o tipo do gasto (remédio, roupa, refeição etc.), a data do gasto, o valor gasto e a forma de pagamento (dinheiro, cartão ou pix). No fim do mês, Vera lista o total de gastos mensais agrupados por tipo de gasto, exibindo o quanto foi gasto em cada tipo de forma de pagamento. Usando UML, identifique a classe, atributos e métodos desse cenário.
- 6) (Rafael Santos, adaptada) Escreva a classe Lampada, considerando que uma lâmpada pode estar acesa ou apagada. Os atributos e métodos da classe devem dar visão do estado da lâmpada, bem como possibilitar sua variação de estado.
- 7) (Rafael Santos) Modifique a resposta da questão anterior para que a classe represente também o número de watts da lâmpada. Escreva um método éEconômica que retorne o valor booleano true se a lâmpada consumir menos de 40 watts e false caso contrário. Não esqueça de atualizar a classe de teste para exercitar o novo método.
- 8) (Rafael Santos) Imagine uma lâmpada que possa ter três estados: apagada, acesa e meia-luz. Usando o modelo Lampada como base, escreva a classe LampadaTresEstados. Que tipo de dado pode ser usado para representar o campo estado?
- 9) (Rafael Santos) Generalize a classe LampadaTresEstados para que ele possa representar uma lâmpada onde a luminosidade pode ser ajustada com qualquer valor entre 0% (apagada) e 100% (acesa). Dica: em vez de operações para possibilitar o ajuste para cada um dos estados, descreva uma operação que receba um valor de ajuste.
- 10) (Rafael Santos) Crie uma classe para representar um time de um esporte qualquer em um campeonato desse esporte. Que dados e operações essa classe deve ter?
- 11) (Rafael Santos, adaptada) Usando UML, projete uma classe para representar uma música para uma *playlist*. Que dados e operações essa classe deve ter? A questão deve ser respondida apenas com o diagrama de classes da UML.

12) Crie uma classe que modele uma pessoa. Toda pessoa deve ter os atributos de nome, idade, peso e altura. Além disso, toda pessoa deve executar os métodos envelhecer, engordar, emagrecer e crescer. Por padrão, a cada ano que a pessoa envelhece, sendo a idade dela menor que 21 anos, ela deve crescer 0,5 cm.

13) (Rafael Santos) Uma das operações que podemos efetuar com datas é a comparação para ver se uma data ocorre antes de outra. O algoritmo para comparação é muito simples, e seus passos estão abaixo. Nesse algoritmo, consideramos que dia1, mês1 e ano1 são os dados da primeira data, e que dia2, mês2 e ano2 são os dados da segunda data.

- Se ano1 < ano2, a primeira data vem antes da segunda.
- Se ano1 > ano2, a primeira data vem depois da segunda.
- Se ano1 == ano2 e mês1 < mês2, a primeira data vem antes da segunda.
- Se ano1 == ano2 e mês1 > mês2, a primeira data vem depois da segunda.
- Se ano1 == ano2 e mês1 == mês2 e dia1 < dia2, a primeira data vem antes da segunda.
- Se ano1 == ano2 e mês1 == mês2 e dia1 > dia2, a primeira data vem depois da segunda.
- Se nenhum desses casos ocorrer, as datas são exatamente iguais.

Crie a classe Data que contenha um método chamado vemAntes que receba como argumento outra instância da classe Data e implemente o algoritmo acima, retornando true se a data encapsulada vier antes da passada como argumento e false caso contrário. Se as datas forem exatamente iguais, o método deve retornar true.

14) (Rafael Santos) Escreva uma classe ModeloDeComputador que encapsule valores que definam a configuração de um microcomputador (tipo de processador, memória RAM, tamanho do disco rígido, tamanho do monitor, por exemplo). Essa classe deve ter um método calculaPreço que calcule o preço do computador como sendo a soma do custo de seus componentes:

- Placa-mãe: R\$800
- Opções de processadores: 600Mhz a R\$700, 800Mhz a R\$830, 933Mhz a R\$910
- Opções de memória: 128, 256, 384 ou 512 Mb, cada 128Mb custa R\$350.
- Opções de disco rígido: 20 Gb a R\$300, 40 Gb a R\$420, 60 Gb a R\$500.
- Opções de monitor: 15 polegadas a R\$320, 17 polegadas a R\$520.

15) (Deitel, adaptada) Segundo a American Heart Association (AHA) a fórmula para calcular a frequência cardíaca máxima por minuto é 220 menos a idade em anos de uma pessoa e que sua frequência cardíaca alvo é um intervalo entre 50-85% da sua frequência cardíaca máxima. Com base nisso, crie uma classe chamada HeartRates. Os atributos da classe devem incluir o nome, sobrenome e a idade da pessoa. A classe também deve incluir um método que calcula e retorna a frequência cardíaca máxima e outro método que calcula e retorna a frequência cardíaca alvo da pessoa. Escreva um programa que instancia um objeto HeartRates, imprime as informações a partir desse objeto — incluindo nome, sobrenome e idade — e calcula e imprime seu intervalo de frequência cardíaca máxima e sua frequência cardíaca alvo.

Desafio: Substituir o atributo idade pela data de nascimento, utilizando a classe Data anteriormente criada, e então incluir um método para calcular e retornar a idade da pessoa.

16) Projete na UML e em seguida crie o código de uma classe para representar um elevador dentro de um prédio. Considere que é possível entrar pessoas no elevador, desde que se respeite sua capacidade, bem como é possível que pessoas saiam do elevador, desde que realmente haja alguém dentro dele. Além disso, o elevador pode subir, desde que não esteja no último andar, e pode descer, desde que não esteja no térreo.

Mais exercícios teóricos e práticos: Exercícios do capítulo 3 do livro de Deitel (10^a Edição) e exercícios dos capítulos 1 e 2 do livro de Rafael Santos.

Observação: Vários exercícios de Deitel consideram o uso da entrada padrão (teclado). Caso você não saiba ou não queira usar o teclado, pode usar valores definidos diretamente no código-fonte.