

Nome do Aluno: \_\_\_\_\_ Data: \_\_\_\_/\_\_\_\_/\_\_\_\_

Prof. Renan Rodrigues de Oliveira

--

## **Trabalho** **Programação Orientada a Objetos**

1. Considerando os conceitos sobre Orientação a Objetos:
  - a) Explique a seguinte afirmação aplicável a projetos de software OO: “Programe para uma interface, não para uma implementação”.
  - b) O que é uma associação de classes? Quais seus principais elementos?
  - c) Conceitue e diferencie os relacionamentos entre classes de agregação e composição.
  - d) Explique a seguinte afirmação na implementação de relacionamentos entre classes: “É de responsabilidade do desenvolvedor prover métodos para vínculo, substituição e/ou remoção da parte”.
  - e) Considere os relacionamentos de agregação e composição. Para cada um destes relacionamentos, crie um modelo que represente uma relação de duas classes com as seguintes multiplicidades: (0..1), (1), (0..\*) e (1..N).
  
2. Considerando a linguagem JAVA, responda as seguintes questões:
  - a) Qual é o propósito da palavra-chave *new*?
  - b) O que é um construtor padrão? Como as variáveis de instância de um objeto são inicializadas se uma classe tiver somente um construtor padrão?
  - c) Explique a diferença de uma variável local e uma variável de instância.
  - d) Explique por que uma classe pode fornecer um método *set* e um método *get* para uma variável de instância.
  - e) O que acontece se se um método contiver uma variável local com o mesmo nome de um dos campos de sua classe?
  - f) Explique os métodos *equals()* e *hashCode()* da linguagem JAVA.
  - g) A API de *Collections* da linguagem JAVA traz a interface *java.util.List*, que especifica o que uma classe deve ser capaz de fazer para ser uma lista. A implementação mais utilizada da interface *List* é a *ArrayList*. Descreva os principais métodos da classe *ArrayList*. Para exemplificar seu uso,

crie uma classe para representar uma cesta de frutas. Não é necessário implementar a classe fruta. Crie apenas o programa principal com uma lista contendo apenas o nome da fruta que está presente na cesta de frutas.

3. Escreva uma classe que represente um país. Um país tem como atributos o seu nome, o nome da capital, sua dimensão em Km<sup>2</sup> e uma lista de países com os quais ele faz fronteira. uma vez feita a modelagem utilizando um diagrama de classe, implemente o programa em Java. Represente a classe e forneça os seguintes construtores e método:
- a) Construtor que inicialize o nome, capital e a dimensão do país;
  - b) Métodos de acesso (*get/set*) para as propriedades indicadas no item (a);
  - c) Um método que permita verificar se dois países são iguais. Dois países são iguais se tiverem o mesmo nome e a mesma capital.
  - d) Um método que defina quais outros países fazem fronteira (note que um país não pode fazer fronteira com ele mesmo);
  - e) Um método que retorne a lista de países que fazem fronteira;
  - f) Um método que receba um outro país como parâmetro e retorne uma lista de vizinhos comuns aos dois países.