



AARE: **Paradigmas de Linguagens de Programação**
Professor: Dr. Ausberto S. Castro V.

Orientação a Objetos

Prova JAVA

Nome Completo:

Data: 31 de maio de 2023

1. **(1 pontos)** Responda detalhadamente as seguintes questões (**exemplifique**):
 - a. Qual é a diferença entre uma classe e um objeto?
 - b. Explique como se define uma classe em Java
 - c. Por que uma classe é uma abstração lógica?
 - d. Como os objetos são criados?
 - e. O que é coleta de lixo e como ela funciona?
2. **(1 pontos)** Desenhar todas as classes, subclasses e superclasses associadas aos seguintes objetos, indicando em cada caso, pelo menos 2 atributos e 2 métodos, e a relação de herança entre classes
 - a. Docente
 - b. Verdura
 - c. Ave
 - d. Computador
 - e. Rede
3. **(01 ponto)** Dar cinco exemplos de cada operação abstrata em orientação a objetos. Ilustre com imagens ou diagramas
4. **(01 ponto)** Escreva um programa em Java que execute as quatro operações aritméticas de números reais
5. **(01 ponto)** Escreva um programa JAVA que calcule a média de cinco notas. O programa deve mostrar o valor da média e indicar se foi aprovado ou reprovado
6. **(01 ponto)** Escreva um programa JAVA que calcule a soma dos inteiros maiores ou iguais a um número inteiro A e menores ou iguais a um inteiro B.
7. **(2 pontos)** Para cada caso da questão 2, escrever um programa JAVA, mostrando as classes, as operações (métodos), a criação de objetos e outras características da Programação Orientada a Objetos.
8. **(02 pontos)** Resolver a Questão 9.10 e 9.11 (pág.363) do livro “Introduction to JAVA Programming”, Y. Daniel Liang, 2014. Não precisa desenhar os diagramas UML. “Implemente a classe” = “Faça um programa em Java”

O professor

***9.10** (Algebra: quadratic equations) Design a class named `QuadraticEquation` for a quadratic equation $ax^2 + bx + c = 0$. The class contains:

- Private data fields `a`, `b`, and `c` that represent three coefficients.
- A constructor for the arguments for `a`, `b`, and `c`.
- Three getter methods for `a`, `b`, and `c`.
- A method named `getDiscriminant()` that returns the discriminant, which is $b^2 - 4ac$.
- The methods named `getRoot1()` and `getRoot2()` for returning two roots of the equation

$$r_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad \text{and} \quad r_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

These methods are useful only if the discriminant is nonnegative. Let these methods return 0 if the discriminant is negative.

Draw the UML diagram for the class and then implement the class. Write a test program that prompts the user to enter values for `a`, `b`, and `c` and displays the result based on the discriminant. If the discriminant is positive, display the two roots. If the discriminant is 0, display the one root. Otherwise, display “The equation has no roots.” See Programming Exercise 3.1 for sample runs.

***9.11** (Algebra: 2×2 linear equations) Design a class named `LinearEquation` for a 2×2 system of linear equations:

$$\begin{array}{rcl} ax + by = e & & \\ cx + dy = f & \Rightarrow & \end{array} \quad x = \frac{ed - bf}{ad - bc} \quad y = \frac{af - ec}{ad - bc}$$

The class contains:

- Private data fields `a`, `b`, `c`, `d`, `e`, and `f`.
- A constructor with the arguments for `a`, `b`, `c`, `d`, `e`, and `f`.
- Six getter methods for `a`, `b`, `c`, `d`, `e`, and `f`.
- A method named `isSolvable()` that returns true if $ad - bc$ is not 0.
- Methods `getX()` and `getY()` that return the solution for the equation.

Draw the UML diagram for the class and then implement the class. Write a test program that prompts the user to enter `a`, `b`, `c`, `d`, `e`, and `f` and displays the result. If $ad - bc$ is 0, report that “The equation has no solution.” See Programming Exercise 3.3 for sample runs.

****9.12** (Geometry: intersecting point) Suppose two line segments intersect. The two end-