



## Curso de Ciência da Computação

### Atividades Avaliativas 4 componentes por grupo

#### Objetivo

Realizar o desenvolvimento das soluções para os problemas e sistemas propostos utilizando o paradigma orientado a objetos.

#### Conceitos básicos:

- Classes e Objetos, Conjunto de Atributos, Comportamentos, Associação e Navegabilidade, Coleções

#### O que deve ser entregue:

- Arquivos contendo os algoritmos em Java e a documentação (JavaDOC).

#### Avaliação:

A avaliação será composta por duas partes:

- Avaliação do material pedido no item “O que deve ser entregue:”
- Arguição INDIVIDUAL dos componentes do grupo, no dia da entrega da atividade.
  - Cada integrante do grupo deverá apresentar alguma parte.
- A NOTA SERÁ INDIVIDUAL, de acordo com os critérios apresentados pelo professor aos alunos da disciplina.

#### Importante:

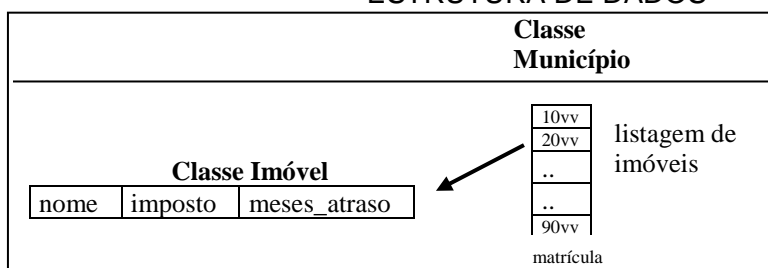
- Data da arguição da atividade: **16 e 20 de Outubro de 2023.**
- Pode utilizar a classe InOut (InOut.java) para obter os dados via Janela de Diálogo.
- Sinta-se livre para propor melhorias, alterações, novas funcionalidades. [Converse com o professor para definir melhor o escopo da implementação.](#)
- Valor desta atividade: 3,0 pontos**

**Atividade 01 – Contexto “IPTU de VV”.** Faça um algoritmo e suas classes para calcular o valor da multa a ser paga ao município por proprietários em atraso, considerando que para cada imóvel conhecemos o nome do proprietário, o imposto a ser pago e os meses de atraso e que os valores das multas são calculados pela seguinte tabela disponível:

TABELA DE IMPOSTOS

Limites Meses		Percentual Multa (%)
0	5	1
6	8	2.3
9	10	3
11	12	5.4
>12		10

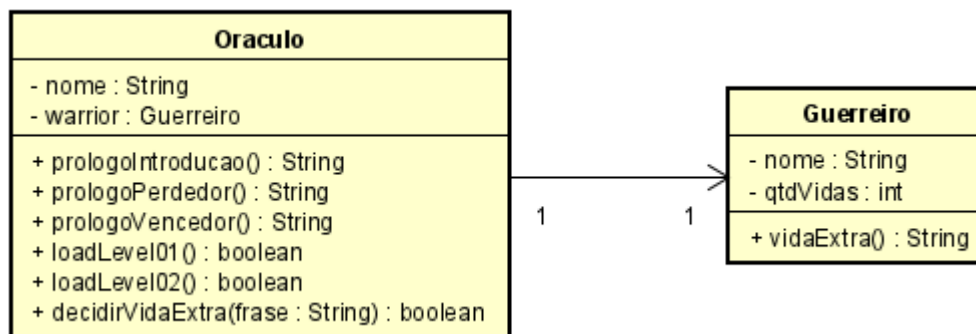
ESTRUTURA DE DADOS



Implemente o sistema. Outros métodos e atributos podem ser incorporados. [Converse com o professor.....](#)

## Atividade 02 – Contexto “Espírito Guerreiro”. Sistema Gamer. Descrição do Sistema:

Escrever um algoritmo em Java que implemente as classes: Estado + Comportamento do diagrama a seguir:



Descrição do Comportamento das classes:

- Método void `setNome(String Nome)`: Este método é para setar o Nome do Oráculo;
- Método int `setVidas()`: Este método faz o sorteio do número de vidas do Guerreiro. A quantidade de vidas deve ser entre [9, 12] vidas;
- Método String `prologoIntroducao()`: Este método exibe na tela um Prólogo (com Nomes) do Oráculo para o Guerreiro + A quantidade de vidas do Guerreiro;
- Método int `loadLevel1()`: Este método faz o sorteio de um Número entre [1, 100]. O Oráculo deve dar uma dica ao Guerreiro: **Segredo é MENOR ou MAIOR que o PALPITE** do número escolhido (obtido via teclado) pelo Guerreiro. **Cada erro do Guerreiro é uma vida a menos!**
- Método int `loadLevel2(int opcao)`: Este método faz um jogo de PAR ou ÍMPAR com o Oráculo. O Guerreiro decide na variável Opção: PAR ou ÍMPAR e o método retorna se o Guerreiro Ganhou ou Perdeu: SE a SOMA deu PAR ou ÍMPAR. O Oráculo e O Guerreiro irão receber números sorteados entre [0, 5].
- Método String `vidaExtra()`: Este método ler (obtido via teclado) um pedido de Misericórdia do Guerreiro por mais UMA(1) Vida;
- Método bool `decidirVidaExtra(String Misericordia)`: Este método recebe o Pedido de Misericórdia do Guerreiro e o Oráculo decide se concede ou não uma (1) Vida Extra. A vida extra será concedida se o Pedido de Misericórdia do Guerreiro tiver mais que cinco palavras.
- Método String `prologoVencedor()`: Este método exibe na tela um Prólogo (com Nomes) do Oráculo para o Guerreiro: VENCEDOR;
- Método String `prologoPerdedor()`: Este método exibe na tela um Prólogo (com Nomes) do Oráculo para o Guerreiro: PERDEDOR;

Na classe Principal (main), faça a instância de um Objeto: Oráculo Master e jogando no MÍNIMO UMA (1) PARTIDA, exibe todo seu Estado + Comportamento, conforme o descritivo do jogo.

**Atividade 03 – Contexto “UVV e-commerce”.** Sistema Pedidos. Descrição do Sistema:

Considere o seguinte diagrama de classes simplificado. Nele, podem ser vistas as classes com algumas sugestões de atributos e métodos.

Implemente o sistema. Outros métodos e atributos devem ser incorporados. [Converse com o professor.....](#)

