



**UNIVERSIDADE PRESIDENTE ANTÔNIO CARLOS - UNIPAC
CONSELHEIRO LAFAIETE - MG**

**ENGENHARIA DA COMPUTAÇÃO
PROGRAMAÇÃO ORIENTADA A OBJETOS I**

Professor: Francesco Silva
francesco.silva@unipaclafaiete.edu.br

**Trabalho - Programação Orientada a Objetos I - JAVA
Engenharia da Computação**

LISTA DE EXERCÍCIOS

Valor: 8 pontos

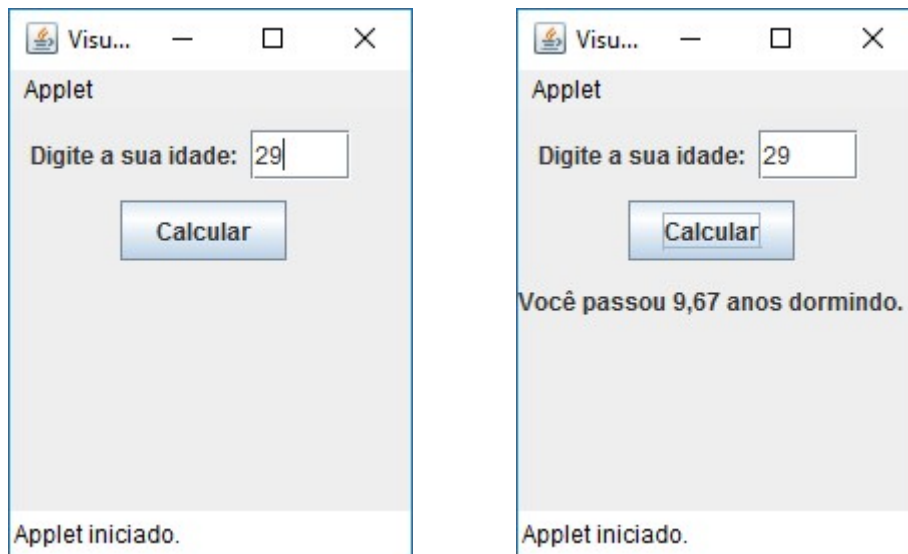
Instruções para entrega do trabalho

- **Data limite para entrega: 23/11/21**
- O aluno deverá criar um único projeto para todos os exercícios da disciplina
- Esse projeto deverá ser armazenado em um repositório privado no Github e o acesso deverá ser liberado para o professor via convite (email para convite francesco.silva@unipac.br)
- O aluno deverá commitar o exercício até a data limite estipulada
- Não é necessário postar o projeto no Blackboard. Apenas commitar o exercício no Git

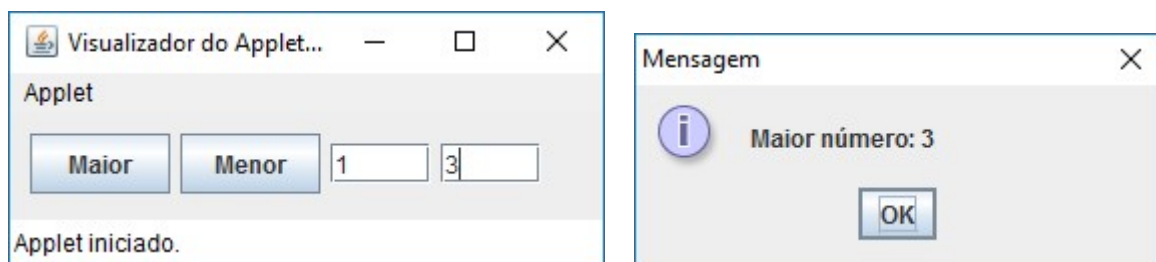
1) Implemente um *aplicativo* de uma calculadora com todos os botões e as quatro principais operações (+ - * /).



2) Crie um *aplicativo* onde o usuário informa em uma caixa de texto sua idade e ao clicar num botão contido na tela, aparecerá em um *label* quantos anos ele passou dormindo (idade / 3). Realize o tratamento de exceção para divisão por zero.



3) Implemente o código de cada botão da tela abaixo, de forma que ao ser clicado o maior, ou menor, número contido nas caixas de texto apareça em um *JOptionPane*.



4) Crie um aplicativo equivalente ao teclado virtual de bancos online. Ele deve conter botões com todos os algarismos, uma caixa de texto para o número da conta, outra para o número da agência e outra para a senha. Quando o usuário clica em um botão o algarismo que correspondente àquele botão é salvo em uma variável e um asterisco aparece na caixa de texto da senha. Crie também um botão para validar a senha. Só será considerada a seguinte conta:

Agência: 1001-9
 Conta: 141414-13
 Senha: 123456



5) Crie um aplicativo para cadastrar álbuns musicais. Tal aplicativo deve receber as seguintes informações: título, artista e número de músicas (estas informações devem ser associadas a uma classe denominada AlbumMusical). Essas informações devem ser recebidas do usuário através de *JTextField*s. Devem ser criados na tela dois botões, Cadastrar e Exibir. O primeiro, insere os dados contidos nos *JTextField*s em um vetor de objetos do tipo AlbumMusical. O segundo lê esse vetor e exibe na tela todos os dados contidos nele.

6) Crie a seguinte classe em Java:

Mes
- nome : String
- salario : double
- gastos : double
- impostos : double
+ getNome() : String
+ setNome(n : String) : void
+ getSalario() : double
+ setSalario(s : double) : void
+ getGastos() : double
+ setGastos(g : double) : void
+ getImpostos() : double
+ setImpostos(i : double) : void
+ saldo() : double

Essa classe faz parte de um sistema que controla os gastos de uma pessoa. Cada instância dessa classe representa um mês desta pessoa. Seus atributos são:

- Nome: nome do mês que o objeto corresponde (janeiro, fevereiro, etc);
- Salário: salário ganho pela pessoa naquele mês;
- Gastos: gastos no mês;
- Impostos: valor total de impostos pagos.

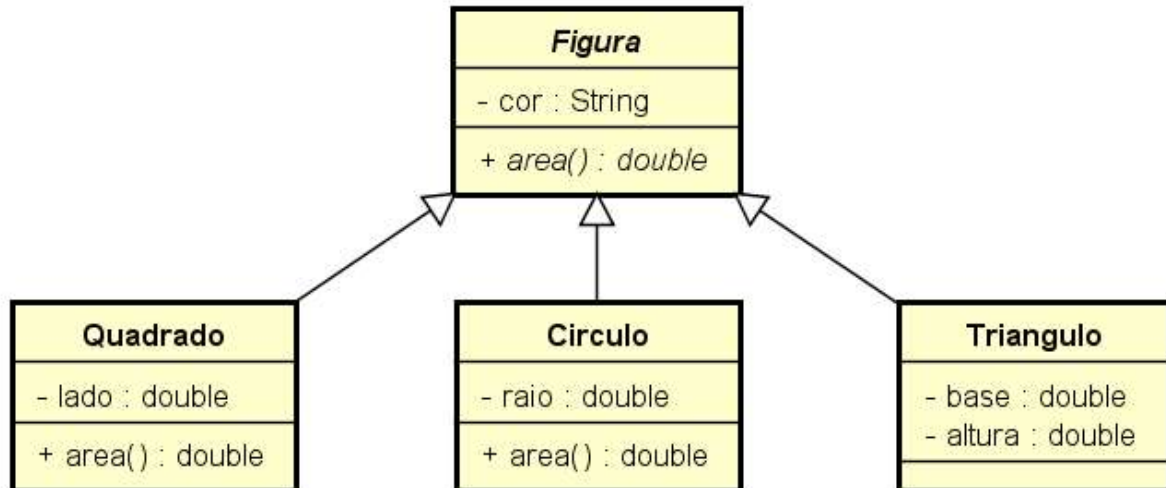
Implemente essa classe. Todos os atributos devem ser encapsulados. O método Saldo retorna o saldo do mês, ou seja, o salário menos os gastos e impostos.

Crie um *aplicativo* que possua um vetor de meses (100 posições). Tal *aplicativo* deve possuir uma interface gráfica para se fazer o cadastro de meses no vetor. Crie também mais três botões na sua interface, com as seguintes funções:

- Cadastrar: insere no vetor os valores digitados pelo usuário
- Exibir: exibe todos os dados do mês que o saldo foi maior;

- Calcular: calcular e exibir a média dos salários recebidos;
- Maior Saldo: exibir o nome e o saldo de todos os meses em um *JOptionPane*.

7) Crie a seguinte estrutura de classes.

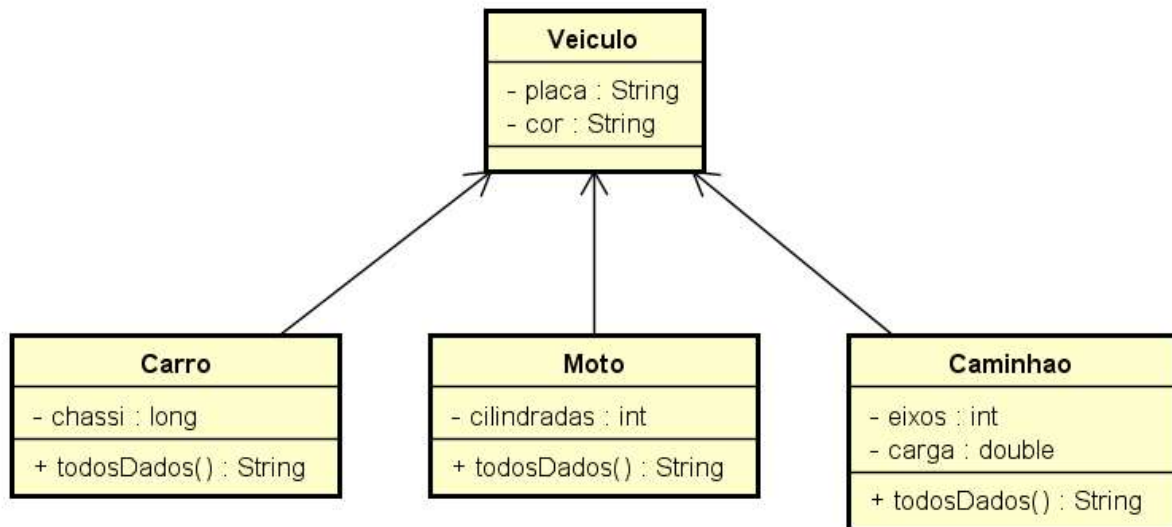


Obs: O método área deve ser polimórfico e a classe figura abstrata.

Crie um *aplicativo* que possua um vetor de 15 posições do tipo *Figura*. Crie uma interface com três botões: um que cadastre Quadrados no vetor, outro que cadastre Círculos e outro que cadastre Triângulos. Os valores de cada objeto devem ser passados pelo usuário através de *JTextFields*. A disposição dos componentes fica a cargo do programador (segue abaixo um exemplo).

Por último crie um botão que ao ser clicado exiba todos os dados da figura que possuir a maior área.

8) Crie a seguinte estrutura de classes:



O método `todosDados` deve retornar uma string contendo todos os atributos do objeto. O construtor de cada classe deve iniciar os objetos com valores nulos.

Crie um *aplicativo* que possua um vetor de objetos do tipo **Veículo**. Faça o usuário preencher tal vetor da mesma forma que o exercício anterior. Ao final, imprima todos os dados de todos os objetos do vetor em um `TextArea`, utilizando para isso o método `todosDados`.

The screenshot shows a Java Applet window titled "Visualizador do Applet: Exer08.class". The applet is titled "Cadastro de Veículos" and prompts the user to "Escolha qual veículo deseja cadastrar:". There are three radio buttons for "Carro", "Moto", and "Caminhão". Below these are input fields for "Cor:", "Placa:", "Chassi:", "Cilindradas:", "Eixos:", and "Carga:". At the bottom of the form are two buttons: "Cadastrar" and "Exibir". Below the form is a large text area displaying the output of the "Exibir" button: "1- Carro Placa: 1234 Cor: azul Chassi: 111". At the very bottom of the window, it says "Applet iniciado."