

aula 1: APRESENTAÇÃO DA **DISCIPLINA**

disciplina: Algoritmos e Programação
III

professora: Sara Guimarães Negreiros



Sara Guimarães Negreiros

Engenheira de Computação pela UFERSA Campus Pau dos Ferros

Entusiasta em computação e tecnologia

GitHub: @guimaraaes

e-mail: sara.negreiros@ufersa.edu.br

1

OBJETIVO DA DISCIPLINA

- Compreender conceitos de programação orientada a objetos com java.
- Java?!
- Objetivo da POO: sistemas de grande porte, provendo modularidade e reusabilidade.

2

MATERIAL DE APOIO



3

EMENTA DA DISCIPLINA

4

PRESENÇA E AVALIAÇÕES

Envio das atividades será
exclusivamente pelo SIGAA

Presença

- Manuscrito para conteúdo discutido em cada aula;
- Precisa ser muito detalhado?

aula N: tema

Avaliações

1ª UNIDADE: Manuscrito (presença) (15%), resolução da lista de exercícios (15%), Prova (35%) e Seminário (35%).

2ª UNIDADE: Manuscrito (presença) (15%) e entrega das funcionalidades requeridas ao longo da unidade para o projeto (85%).

3ª UNIDADE: Manuscrito (presença) (15%) e entrega das funcionalidades requeridas ao longo da unidade para o projeto (85%).

PONTUAÇÃO EXTRA (1.0 ponto em cada unidade): Participação nas aulas.

5

ENCONTROS E HORÁRIO DE DÚVIDAS

```
graph TD; A[Aulas] --> B[assíncronas: gravadas e disponíveis no YouTube]; A --> C[Síncronas: dúvidas e discussões];
```

Aulas

assíncronas: gravadas e
disponíveis no YouTube

Síncronas: dúvidas e
discussões

Horário de dúvidas: 3N1234



DÚVIDAS?

6

CONTEXTO DE PROGRAMAÇÃO

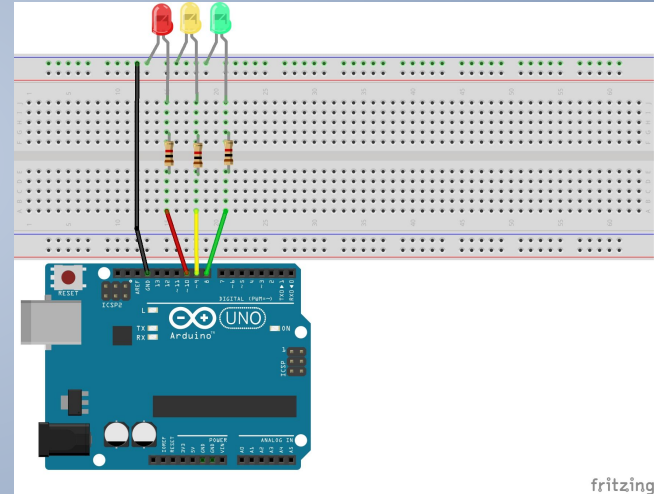
CONTEXTO DE PROGRAMAÇÃO

- Qual a motivação de terem desenvolvidos computadores?
- Se os computadores realizam apenas operações matemáticas, qual a grande vantagem disso?
- $1.8 \text{ GHz} = 1.8 * 10^9 * 1/T = \mathbf{1\ 800\ 000\ 000}$ * instruções por segundo

```
sara@notebook:~$ lscpu
Arquitetura:                x86_64
Modo(s) operacional da CPU: 32-bit, 64-bit
Ordem dos bytes:            Little Endian
Address sizes:              39 bits physical, 48 bits virtual
CPU(s):                     8
Lista de CPU(s) on-line:    0-7
Thread(s) per núcleo:       2
Núcleo(s) por soquete:      4
Soquete(s):                 1
Nó(s) de NUMA:              1
ID de fornecedor:           GenuineIntel
Família da CPU:             6
Modelo:                     142
Nome do modelo:             Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz
```

- Qual a diferença entre algoritmo, código, pseudocódigo?
- Quais são os exemplos de linguagem de programação? Os computadores compreendem apenas uma?
- Baixo nível vs Alto nível.

Funcionamento de um semáforo



7

INTRODUÇÃO A POO

MODELOS

- O que caracteriza algo como um modelo?
- “representação, em escala reduzida, de objeto, obra de arquitetura etc. a ser reproduzida em dimensões normais.”
- O que é um modelo matemático ou um modelo de ensino?

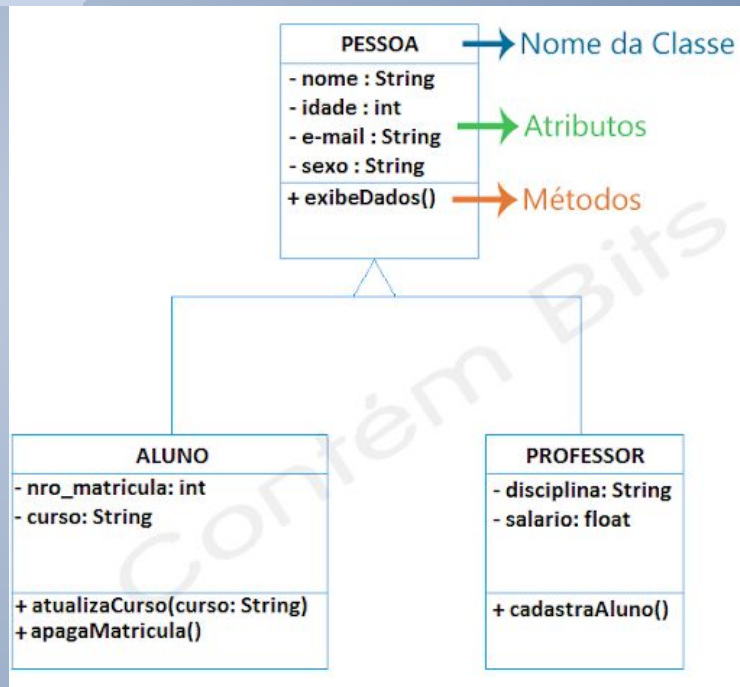
MODELOS

- Dados;
- Operações;
- Modelo dentro de modelo?
- Kg refeição:
 - ▷ kg carne,
 - ▷ kg arroz,
 - ▷ kg feijão

Restaurante Caseiro Hipotético		
Mesa 1 <input type="checkbox"/> kg refeição <input type="checkbox"/> sobremesa <input type="checkbox"/> refriger. 2 L. <input type="checkbox"/> refriger. 600mL. <input type="checkbox"/> refriger. lata <input type="checkbox"/> cerveja	Mesa 2 <input type="checkbox"/> kg refeição <input type="checkbox"/> sobremesa <input type="checkbox"/> refriger. 2 L. <input type="checkbox"/> refriger. 600mL. <input type="checkbox"/> refriger. lata <input type="checkbox"/> cerveja	Mesa 3 <input type="checkbox"/> kg refeição <input type="checkbox"/> sobremesa <input type="checkbox"/> refriger. 2 L. <input type="checkbox"/> refriger. 600mL. <input type="checkbox"/> refriger. lata <input type="checkbox"/> cerveja
Mesa 4 <input type="checkbox"/> kg refeição <input type="checkbox"/> sobremesa <input type="checkbox"/> refriger. 2 L. <input type="checkbox"/> refriger. 600mL. <input type="checkbox"/> refriger. lata <input type="checkbox"/> cerveja	Mesa 5 <input type="checkbox"/> kg refeição <input type="checkbox"/> sobremesa <input type="checkbox"/> refriger. 2 L. <input type="checkbox"/> refriger. 600mL. <input type="checkbox"/> refriger. lata <input type="checkbox"/> cerveja	Mesa 6 <input type="checkbox"/> kg refeição <input type="checkbox"/> sobremesa <input type="checkbox"/> refriger. 2 L. <input type="checkbox"/> refriger. 600mL. <input type="checkbox"/> refriger. lata <input type="checkbox"/> cerveja

MODELOS E POO

- O que seria um super modelo?
- Quais as vantagens e desvantagens?
- A POO se baseia em modelos para definir classes, atributos e métodos.
- Modelos: dados e regras de negócio.
- Porque PO**Objetos**?



ENCAPSULAMENTO

- Mundialmente a proteção de dados em sistemas de software é um requisito básico.
- Encapsular dados e métodos é definir as condições de visibilidade.
- Público, privado ou protegido.



CLASSES

- Representação abstrata dos modelos.
- Objetos: instâncias de classes (materialização).
- Os dados possuem um tipo (inteiro, float, booleano, string).
- Os métodos podem receber argumentos e/ou retornar valores.

LÂMPADA

Lampada
- estadoDaLampada
- acende()
- apaga()
- mostraEstado()

```
modelo Lampada // representa uma lâmpada em uso
início do modelo
    dado estadoDaLampada; // indica se está ligada ou não

    operação acende() // acende a lâmpada
        início
            estadoDaLampada = aceso;
        fim

    operação apaga() // apaga a lâmpada
        início
            estadoDaLampada = apagado;
        fim

    operação mostraEstado() // mostra o estado da lâmpada
        início
            se (estadoDaLampada == aceso)
                imprime "A lâmpada está acesa";
            senão
                imprime "A lâmpada está apagada";
            fim
        fim

fim do modelo
```


9

ALGUMAS CONSIDERAÇÕES

- Qualquer sistema pode ser definido e elaborado com POO, cabe ao programador realizar uma boa análise dos requisitos e saber conversar com o cliente.
- Seguir boas práticas de código é fundamental.

10

FERRAMENTA DE TRABALHO

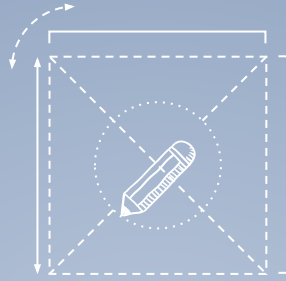
Nosso primeiro programa



eclipse



EXERCÍCIOS



1. Definir o diagrama de classes para um sistema de estoque e um sistema de escola.
2. Explique, com exemplos, por que seria complicado usar um “super modelo” que representaria todos os dados de uma pessoa.



DÚVIDAS?

sara.negreiros@ufersa.edu.br