

# Orientação a Objetos I

## Aula 1 – Conceitos introdutórios

Prof. Humberto Martins Beneduzzi  
humberto.beneduzzi@ifpr.edu.br

# Agenda

- Apresentação
- Objetivos da disciplina
- Metodologia de avaliação
- Cronograma da disciplina
- Conceitos introdutórios de O.O.
- Atividade prática

# Objetivos

- Propiciar ao aluno a compreensão dos principais aspectos do paradigma de desenvolvimento de software orientado a objetos.
- Realizar a implementação destes conceitos em uma linguagem de programação.

# Ementa

- O paradigma Orientado a Objetos.
- Conceitos fundamentais da Orientação a Objetos: Classe, Objeto, Atributo, Método, Encapsulamento, Associação (Agregação e Composição), Herança, Polimorfismo, Interfaces.
- Aplicação dos conceitos da Orientação a Objetos em uma linguagem de programação.

# História da OO

- Início com **Simula (1966)**, que era uma extensão da família de linguagens de programação de alto nível utilizadas principalmente para aplicações científicas *ALGOL (ALGOritmic Language)*.
  - Introduziu os conceitos de classes e herança;
  - Na linguagem, uma classe é um módulo que engloba as estruturas e comportamento comuns aos objetos;
  - Voltada para simulações: os sistemas são modelados pela interação de vários objetos distintos.

# História da OO

- **Smalltalk** (década de 70):
  - Autores: Alan Kay, Dan Ingalls, Adele Goldberg;
  - Em Smalltalk tudo é objeto: números, classes, métodos, blocos de código, entre outros;
  - Linguagem considerada puramente orientada a objetos.
- **C++** (surge em 1979. Nome original: *C with Classes*)
  - Autor: Bjarne Stroustrup;
  - Uma das linguagens comerciais mais populares, e altamente difundida na academia graças a seu desempenho.

# História da OO

- JAVA (1995)
  - Autores: James Gosling, Sun Microsystems;
  - Ao contrário de linguagens de programação convencionais que são compiladas para código nativo, a linguagem Java é compilada para um bytecode, que por sua vez é interpretado por uma máquina virtual (Java Virtual Machine - JVM)

# Paradigma

**Segundo o filósofo Thomas Kuhn:**

**Os paradigmas são conjuntos de regras e regulamentos que estabelecem limites e sinalizam como resolver problemas dentro desses limites.**

**Os paradigmas agem como filtros dos dados que chegam à nossa mente.**



# Contextualização

## **Paradigma Estruturado**

- Fornece um conjunto de estruturas de controle, que permitem descrever a lógica necessária para a construção de algoritmos. Procura-se quebrar um problema complexo em problemas menores.

## **Paradigma Orientado a Objetos**

- Na visão orientada a objetos, o mundo é composto por um conjunto de objetos que possuem atributos e operações, e interagem entre si.
- O modelo orientado a objetos permite modelar entidades do mundo real, de forma que estas entidades possam ser representadas no contexto de um software.

# Classe

- É a representação de uma categoria (classe) de objetos do mundo real, contendo apenas as informações (atributos e métodos) importantes no contexto do sistema.
- É composta de atributos e métodos.
- Atributo = característica
- Método = operação ou comportamento (algo que o objeto pode ou permite fazer)

# Classe

Exemplo:

- **Classe:** Pessoa
- **Atributos:** nome, altura, dtNascimento
- **Métodos:** andar, falar, dormir
- Uma classe é um modelo para a criação de objetos de um determinado tipo.

# Objeto

- É uma instância de uma classe.
- É uma entidade baseada na classe, porém independente, que possui os métodos e atributos definidos na classe.
- O valor dos atributos e a execução dos métodos é independente em cada objeto.
- Exemplo: Os objetos João e Pedro pertencem à classe Pessoa.

# Exercícios

- Defina os atributos e métodos para uma classe Televisor
- Defina os atributos e métodos para uma classe Telefone

# Abstração

- **Abstração** é o processo mental que consiste em escolher ou isolar um aspecto determinado de um estado de coisas relativamente complexo, a fim de simplificar a sua avaliação, classificação ou para permitir a comunicação [verbalizar, apresentar textualmente, codificar, ...] do mesmo.  
Houaiss, 2006

# Abstração

- No contexto da OO:
  - É a capacidade de extrair as principais características e comportamentos que definem uma determinada categoria (ou classe) de objetos do mundo real, ignorando as informações irrelevantes para o contexto do sistema.

# Abstração

Exemplo:

- Quantas características uma cadeira tem?
  - (listar ao menos 10 atributos de uma cadeira, e quantas operações conseguir)



# Abstração

Exemplo:

- Quantas características uma cadeira tem?
  - **Atributos:** cor, preço, marca, corPredominante, corAssento, alturaAssento, alturaEncosto, tamanhoEncosto, tipoPes, anguloEncosto
  - **Métodos:** girar, ajustarAlturaAssento, ajustarAlturaEncosto, ajustarAnguloEncosto
- Quais destas características o meu sistema precisa ter?
- Através do processo de abstração realizamos a definição das classes de um sistema.

# Exercícios

Exercícios:

1- Defina uma classe Carro para o sistema de uma revenda de automóveis;

# Exercícios

Exercícios:

- 1- Defina uma classe Carro para o sistema de uma revenda de automóveis;
- 2- Defina uma classe Carro para o sistema de um lava car;

# Exercícios

Exercícios:

- 1- Defina uma classe Carro para o sistema de uma revenda de automóveis;
- 2- Defina uma classe Carro para o sistema de um lava car;
- 3- Defina uma classe Carro para o sistema de uma montadora de automóveis.

# Exercícios

Exercícios:

- 1- Defina uma classe Carro para o sistema de uma revenda de automóveis;
  - 2- Defina uma classe Carro para o sistema de um lava car;
  - 3- Defina uma classe Carro para o sistema de uma montadora de automóveis.
- Compare as diferenças entre as classes definidas