UNIP

UNIVERSIDADE PAULISTA

Paradigmas de Linguagens

Paradigma Orientado a Objetos

Professora Sheila Cáceres

Conteúdo

- Histórico de OO
- Conceitos Básicos de OO
- Vantagens de OO
- Linguagens OO
- Reflexão

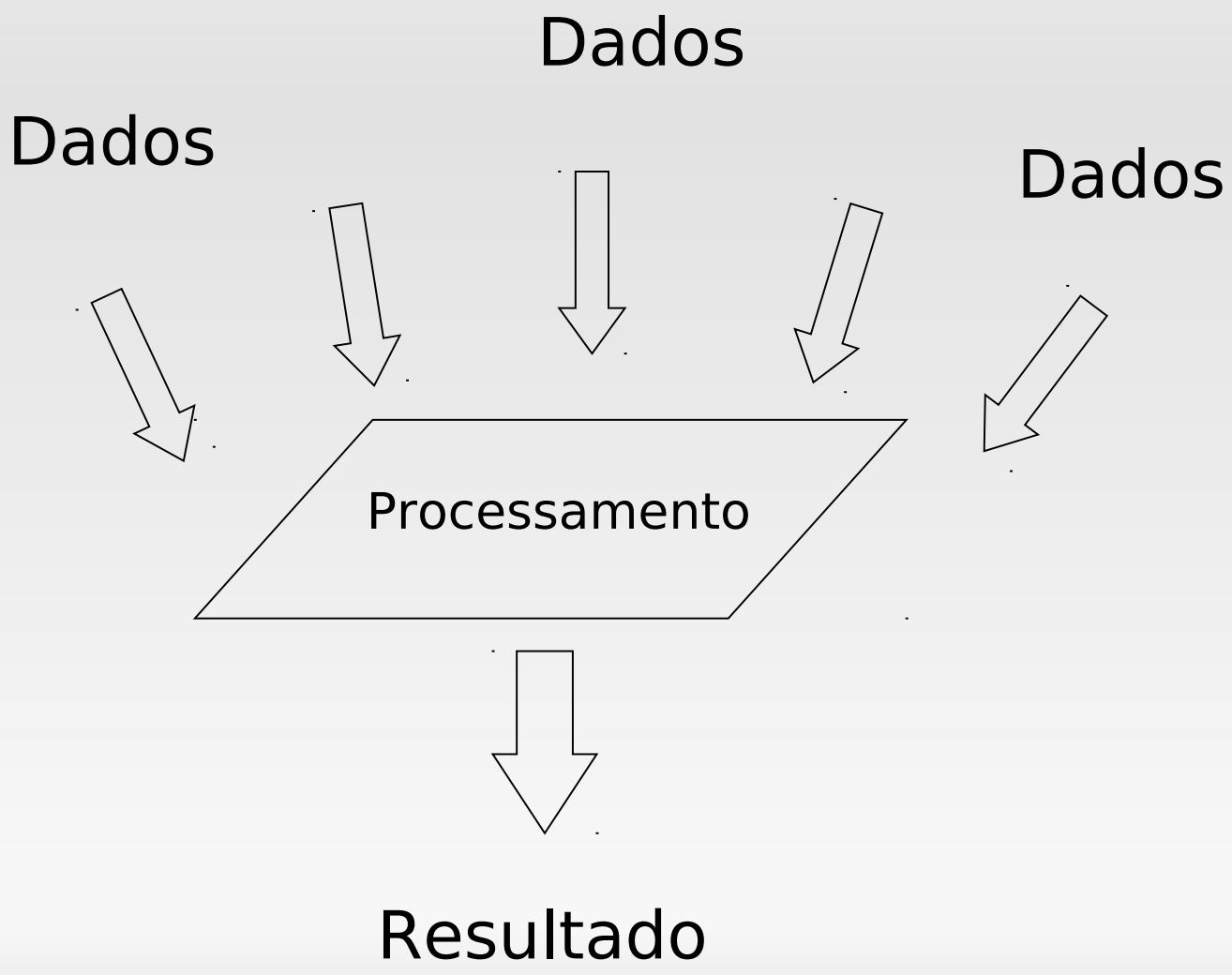
Histórico de OO

- A OO surgiu no final da década de 60, quando dois cientistas dinamarqueses criaram a linguagem Simula (Simulation Language) → Simula-67 introduzindo os conceitos de classe e herança.
- O termo Programação Orientada a Objetos (POO) é introduzido com a linguagem Smalltalk (1983).
- FINS DOS ANOS 80 ⇨ Paradigma de Orientação a Objetos começo a ser considerada uma abordagem poderosa e prática para o desenvolvimento de software
- Surgiram linguagens híbridas:
  - C++ (1986), Object-Pascal (1986)

Histórico de OO

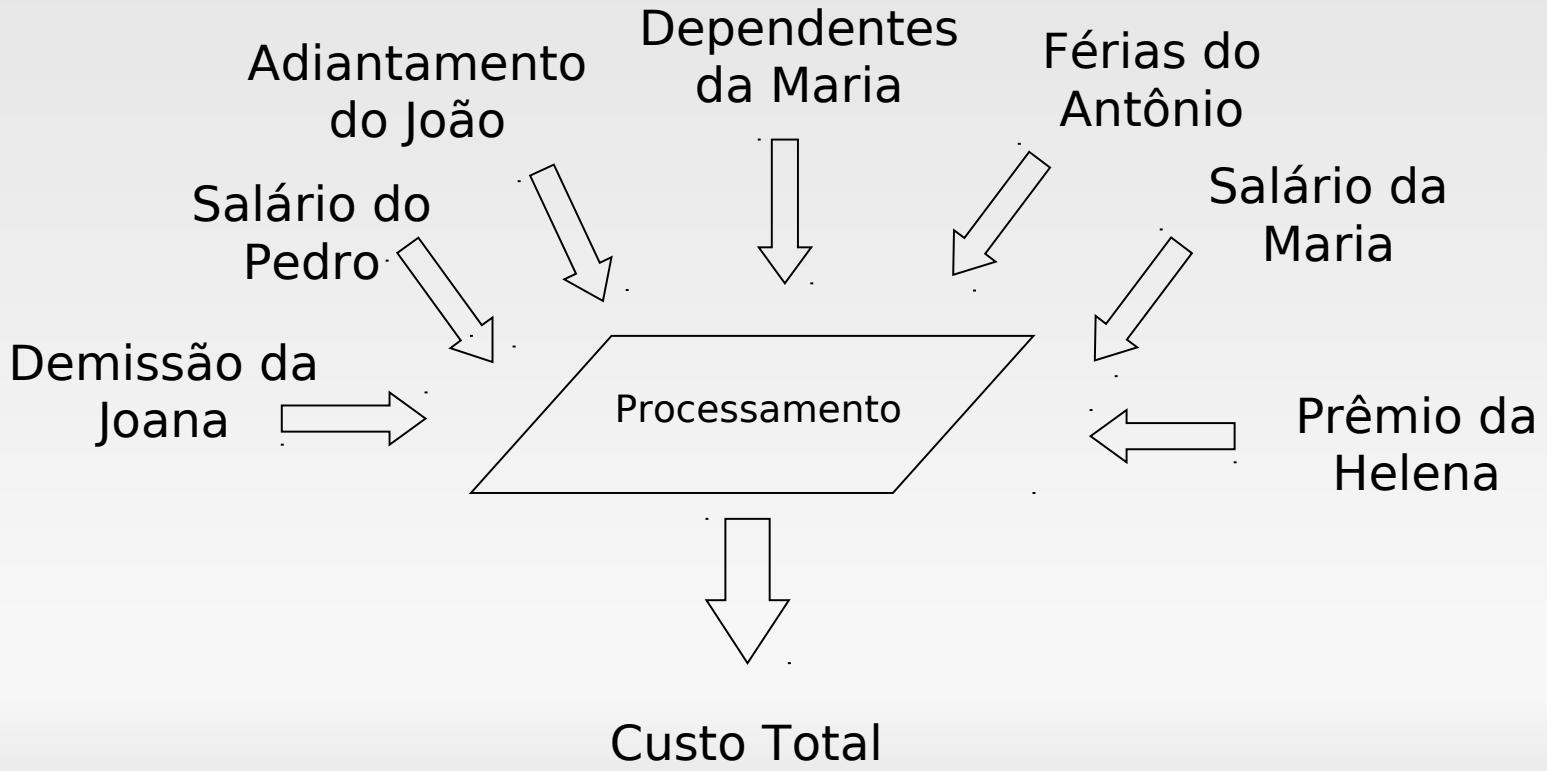
- A medida que o tempo passava, os sistemas ficavam cada vez mais complexos.
- A engenharia de software surgiu para organizar esforços no desenvolvimento de software e metodologias & ambientes para o mesmo.
- Assim, muitas ferramentas, metodologias e ambientes dependem do Paradigma (Modelo, filosofia) de Desenvolvimento.
- Como vimos, o paradigma imperativo se baseia em um modelo entrada-processamento-saída

Paradigma Imperativo



Paradigma Imperativo

Exemplo: Calcular a folha de pagamento de um departamento



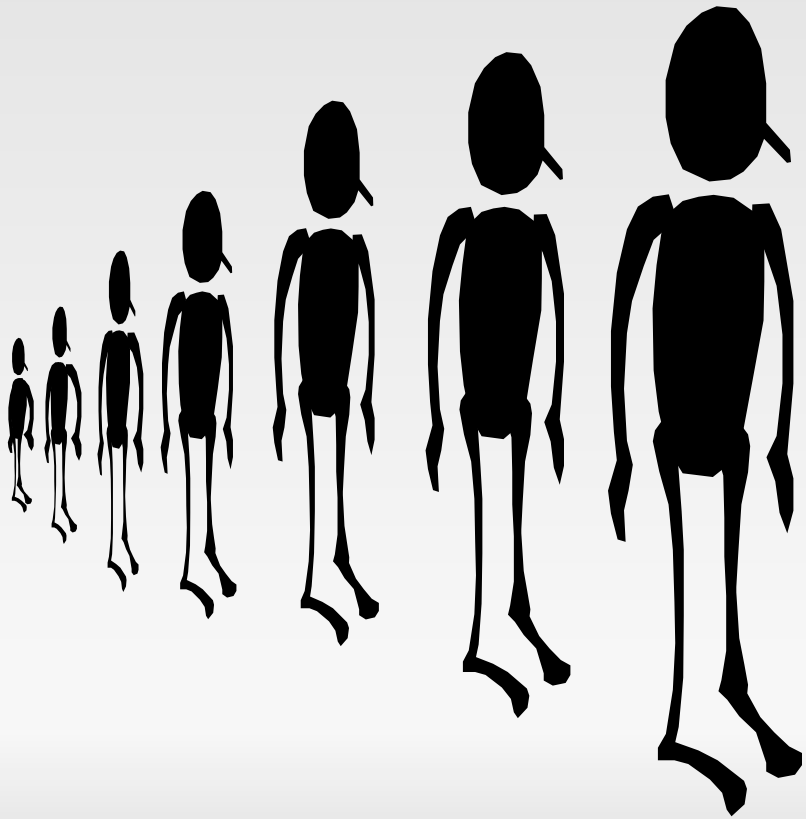
Então o Paradigma Imperativo não é mais suficiente.

Paradigma Orientação a Objetos (OO)

- É uma abordagem de programação mais intuitiva.
- Pressupõe que o mundo é composto por *objetos.*
- Podemos representar no computador objetos análogos aos objetos existente no mundo real.
- Os sistemas são modelados como um conjunto de objetos que interagem entre eles.

Paradigma OO

- **Exemplo**: Calcular a folha de pagamento de um departamento

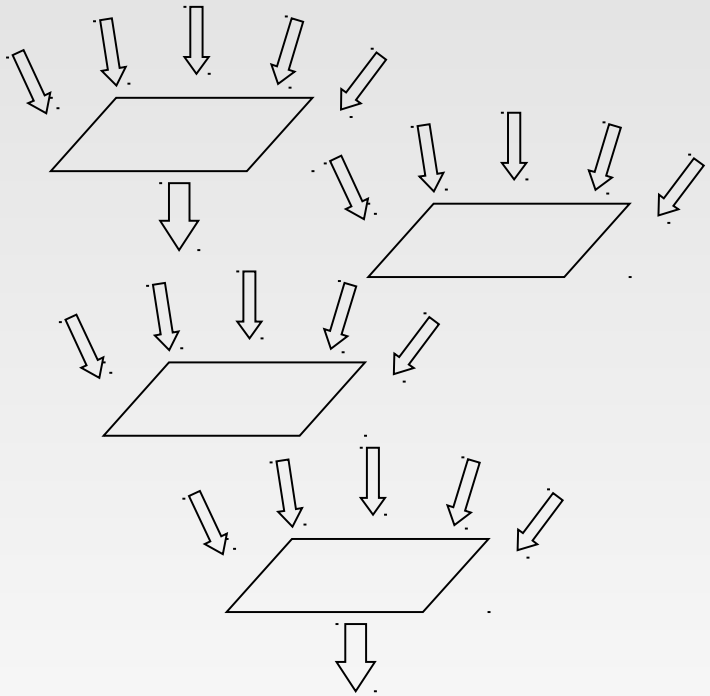


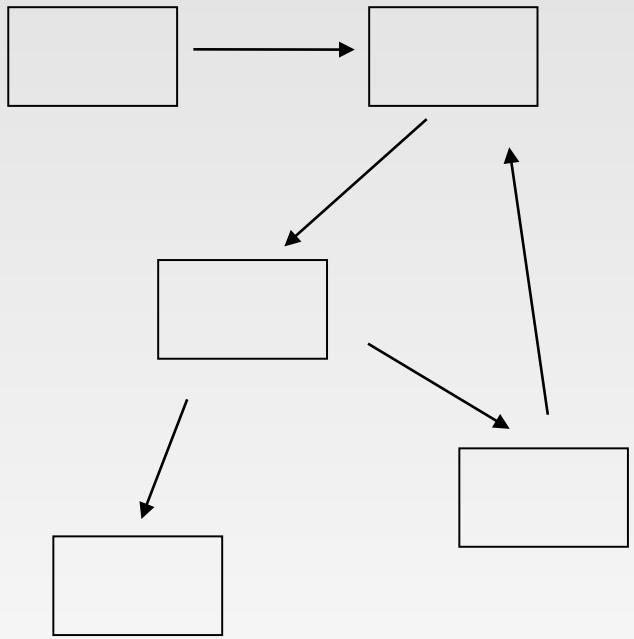
"Fulano, quanto eu te devo?"

Custo total =

Σ Fulanos

Imperativo *vs* OO





Paradigma Orientado a Objetos

Em resumo

- No enfoque de OO, as unidades básicas são os objetos que trocam mensagens entre si.
- Essas mensagens resultam na ativação de métodos, os quais realizam as ações necessárias (serviços solicitados pelas mensagens).
- As classes agrupam os objetos com o mesmo comportamento (objetos que respondem as mesmas mensagens).

Conceitos Básicos

Objetos e Classes

Objeto

- Objeto é uma entidade abstraída do mundo real.
- Um objeto pode ser construído, executar ações, etc.

- Cada objeto é um elemento abstrato responsável pelo seu estado (definido pelos atributos) e faz as transformações sobre tal estado mediante um conjunto fixo de regras de comportamento (métodos).

Objetos:

- Tudo em OO é OBJETO
- Definição (mundo do sofware)
  - “Qualquer coisa, real ou abstrata, a respeito da qual armazenamos dados e métodos que os manipulam” Martin, Odell (1995)
- Abstração de uma entidade do mundo real de modo que essa entidade possue várias características

Classes

- No mundo real, diferentes objetos desempenham o mesmo papel

Automóveis

- A estrutura e o comportamento comuns dos objetos podem ser agrupados

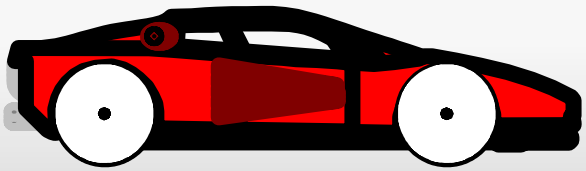
Classes

- Permitem definir um tipo de objetos.
- Todo objeto é uma **instância** de uma Classe.
- Os objetos representados por determinada classe diferenciam-se entre si pelos valores de seus atributos.
- As classes definem então um conjunto (agrupamento) de objetos similares pois possuem propriedades semelhantes (**ATRIBUTOS**), o mesmo comportamento (**MÉTODOS**), os mesmos relacionamentos com outros objetos e a mesma semântica.

Classes

- Modelo que descreve a estrutura e o comportamento de objetos comuns = *Classe*
- Objetos que se comportam da maneira especificada pela classe são ditos *Instâncias* dessa classe
- Ex

Ferrari do Luciano



é um Instância da Classe Automóvel

Exemplo

Casa

(Classe)

Portas
Quartos
Salas
Localização
Cozinha
Telhado

Reformar
Limpar
Pintar
Mobilhar

Objetos

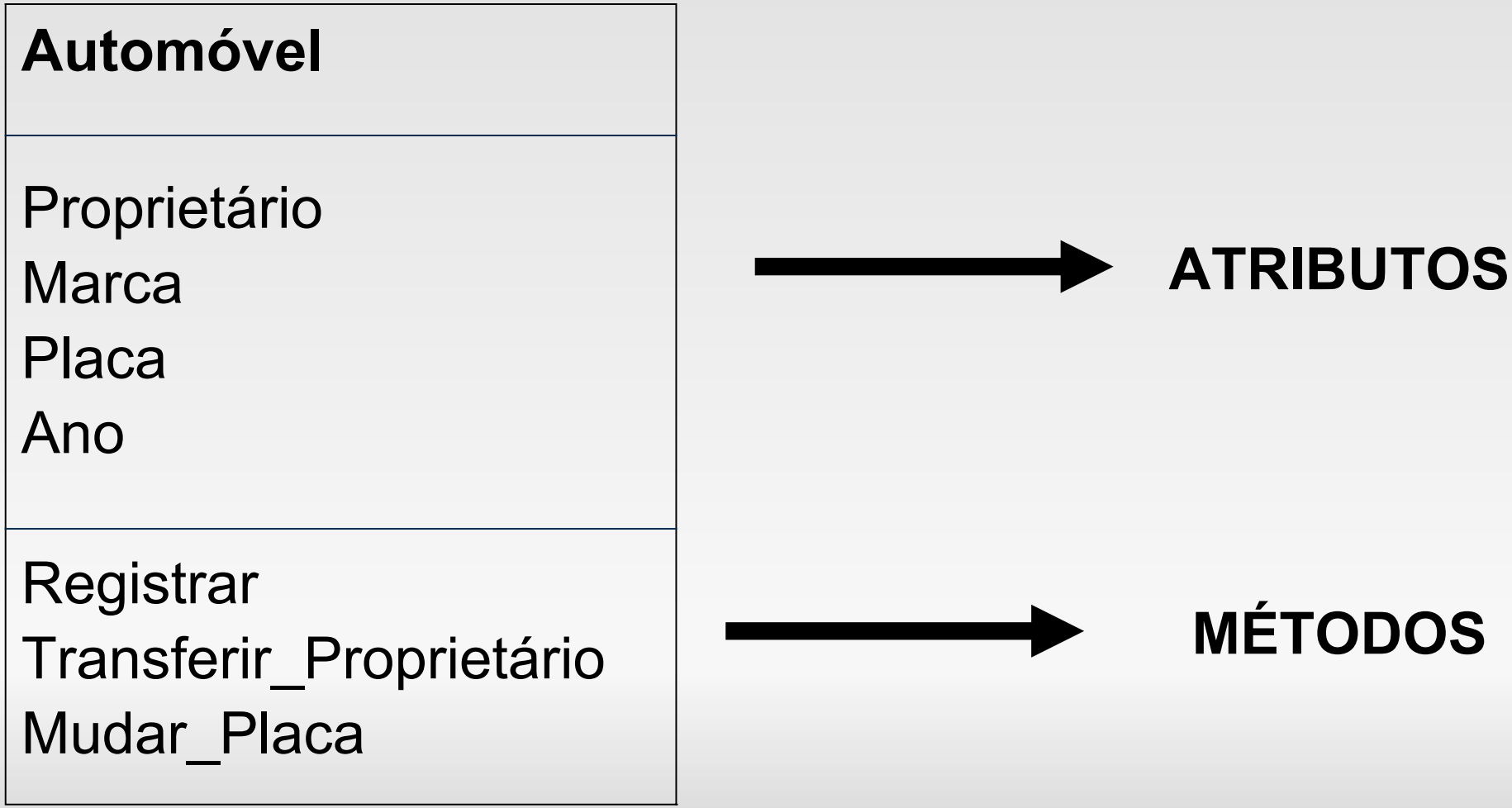
(Instancias da classe)

Casa do Presidente

Casa da Maria

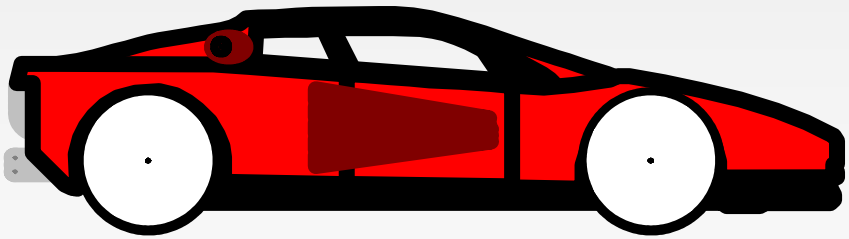
Casa Amarela

Atributos e Métodos



Estado do Objeto (Atributos)

- Conjunto de propriedades de um objeto associadas a seus valores correntes
  - Cada objeto possue seu próprio conjunto de atributos
  - Representam as informações, ou dados que caracterizam um objeto
- Ex:



Propriedades:

- Cor = Vermelha
- Ano = 2001
- Proprietario =Luciano
- Placa = EBTH 2435

Estado do Objeto (Atributos)

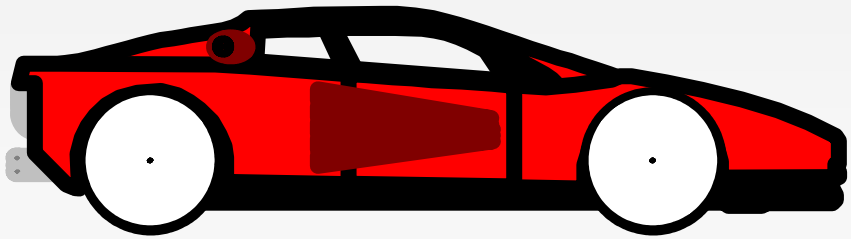
- Os valores das propriedades de um objeto definem seu **Estado**.

	Maria	José	João
• idade	31	28	43
• endereço	Rua xx	Rua yy	Av zz
• sexo	Fem	Masc	Masc
• etc			

Maria José João

Comportamento do Objeto (Métodos)

- Conjunto de serviços ou operações que outros objetos podem requisitar
- Representa como o objeto reage às *Mensagens* a ele enviadas
  - Métodos são invocados por Mensagens
  - Cada objeto possue seu próprio conjunto de métodos
- Ex

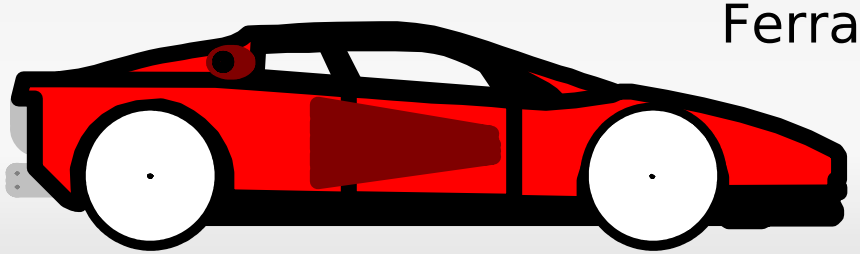


Operações

- Acelerar ( )
- Frear ( )
- Acender Faróis ( )
- Virar a Direita ( )

Identidade Única

- Objetos têm existência própria
- São distintos mesmo se seus estados e comportamento forem iguais
- Identificador que permite referência inequívoca
- Ex



Ferrari do Luciano

Exemplo

- Ex

Classe Automóvel:

Propriedades:

Cor, Ano, Velocidade, Combustível

Operações:

Acelerar( ), Frear( ), Acender Faróis( ), Virar a Direita( )

Conceitos Básicos

- Três elementos chaves de OO são:
  - Encapsulamento
  - Herança
  - Polimorfismo

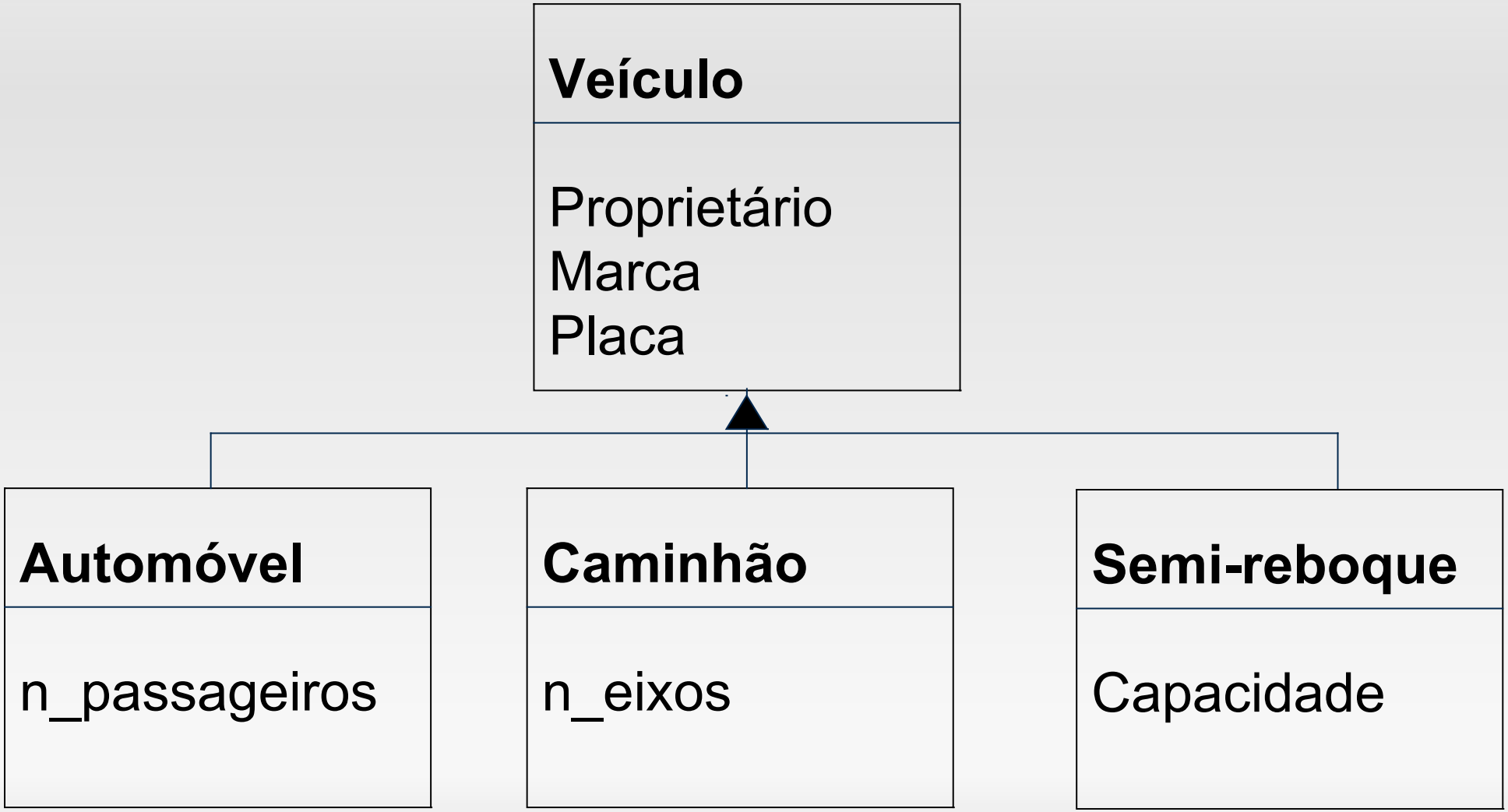
Encapsulamento

- Objetos encapsulam seus atributos;
- Propriedade segundo a qual os atributos de uma classe são acessíveis apenas pelos métodos da própria classe;
- Outras classes só podem acessar os atributos de uma classe invocando os métodos públicos;
- Restringe a visibilidade do objeto mas facilita o reuso
- Os DADOS e os MÉTODOS são empacotados sob um nome e podem ser reusados como uma especificação ou componente de programa.

Herança:

- É o mecanismo pelo qual uma subclasse herda todas as propriedades da superclasse e acrescenta suas próprias e exclusivas características.
- As propriedades da superclasse não precisam ser repetidas em cada subclasse.
- Por exemplo, JanelaRolante e JanelaFixa são subclasses de Janela. Elas herdam as propriedades de Janela, como uma região visível na tela. JanelaRolante acrescenta uma barra de paginação e um afastamento.

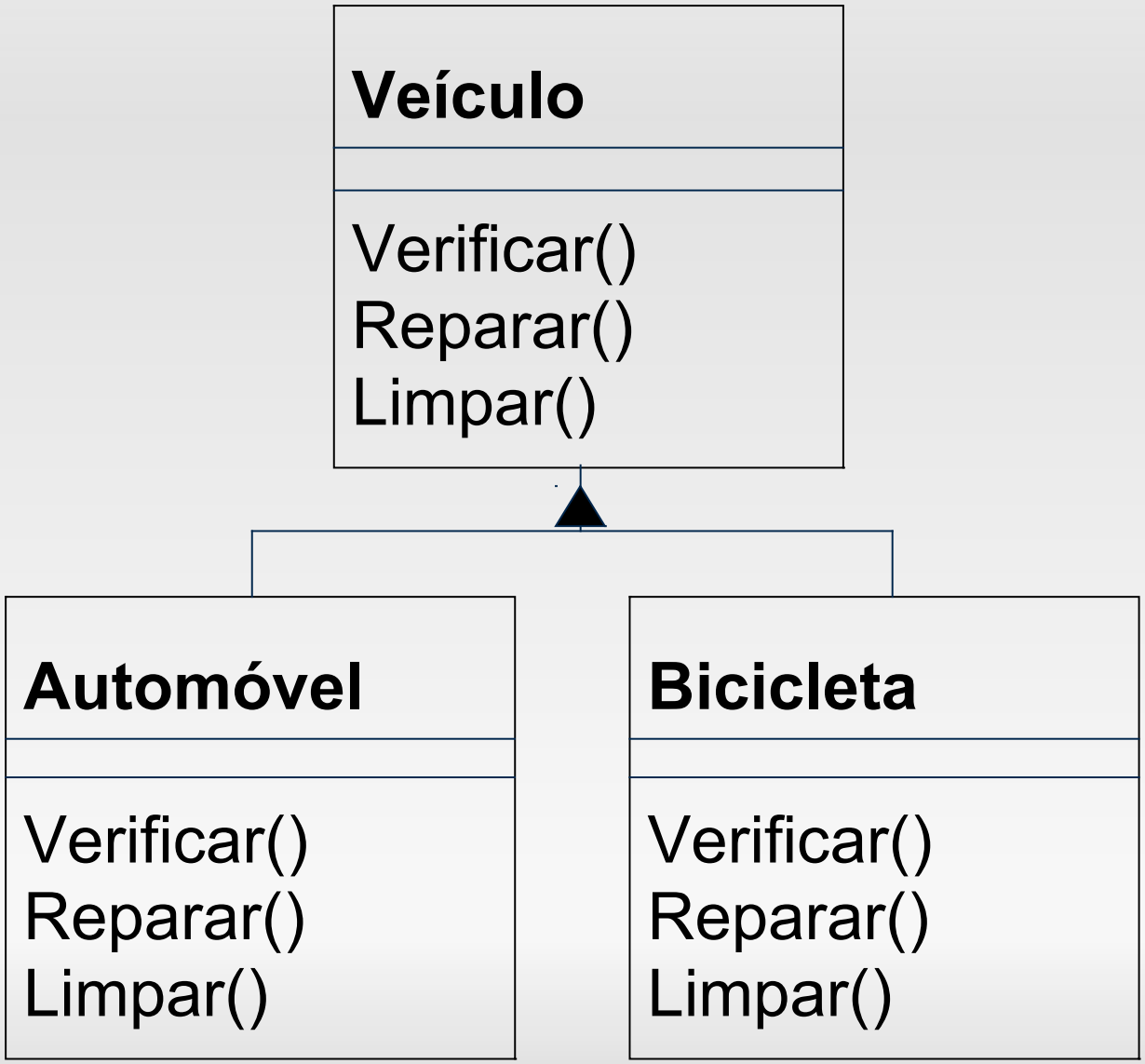
Herança


Veículo
Proprietário
Marca
Placa
Automóvel
n_passageiros
Caminhão
n_eixos
Semi-reboque
Capacidade

Polimorfismo

- Geralmente representa a qualidade ou estado de um objeto ser capaz de assumir diferentes formas.
- Mais especificamente, propriedade segundo o qual vários métodos podem existir com o mesmo nome.
- Ao receber uma mensagem para efetuar uma Operação, é o objeto quem determina como a operação deve ser efetuada;
- Permite a criação de várias classes com interfaces idênticas, porém objetos e implementações diferentes.
- Exemplos:
  - O operador “+” pode ser usado com inteiros, pontos-flutuantes ou strings.
  - A operação mover pode atuar diferentemente nas classes Janela e PeçadeXadrez.

Polimorfismo



Analogia

Analogia entre conceitos no paradigma orientado a objeto e no paradigma imperativo

Linguagens Orientadas a Objetos	**Linguaguens Tradicionais**
Objeto	Valor
Mensagem	Chamada de Procedimento
Método	Função ou Procedimento

Reflexão

- A tecnologia de OO é bastante recente e “veio para ficar”
- OO impõe qualidade, produtividade e profissionalismo na construção de sistemas
- Existem métodos, técnicas e ferramentas de software OO que acompanham o processo de desenvolvimento do software desde a análise até a implementação

Vantagens de OO

- Abstração de dados: os detalhes referentes às representações das classes serão visíveis apenas a seus atributos;
- Compatibilidade: as heurísticas para a construção das classes e suas interfaces levam a componentes de software que são fáceis de se combinar;
- Flexibilidade: as classes delimitam-se em unidades naturais para a alocação de tarefas de desenvolvimento de software;

Vantagens de OO

- Reutilização: o encapsulamento dos métodos e representação dos dados para a construção de classes facilitam o desenvolvimento de software reutilizável, auxiliando na produtividade de sistemas;
- Extensibilidade: facilidade de estender o software devido a duas razões:
  - herança: novas classes são construídas a partir das que já existem;
  - as classes formam uma estrutura fracamente acoplada o que facilita alterações;
- Manutenibilidade: a modularização natural em classes facilita a realização de alterações no software.

Vantagens de OO

- Melhoria de comunicação entre desenvolvedores e clientes;
- Redução da quantidade de erros no sistema, diminuindo o tempo nas etapas de codificação e teste;
- Maior dedicação à fase de análise, preocupando-se com a essência do sistema;
- Mesma notação é utilizada desde a fase de análise até a implementação.

Linguagens OO

- Existem diversas linguagens OO, tais como:
  - Smalltalk (1972)
  - Ada (1983)
  - Eiffel (~1985)
  - Object Pascal (1986)
  - Common Lisp (1986)
  - C++ (~1989)
  - Java

Referências Bibliográficas

O material para a realização desta apresentação foi extraido de:

* Tucker, Linguagens de Programação, principios e paradigmas.

* Slides do Prof. Ricardo Satoshi Oyakawa

* Principios de Linguagens de programação, Ana Cristina Vieira de Melo, Flavio Soares