

cin.ufpe.br



Centro de Informática

U • F • P • E



UNIVERSIDADE FEDERAL DE PERNAMBUCO

Paradigma Orientado a Objeto

Visão Geral

Por:

Robson do Nascimento Fidalgo
rdnf@cin.ufpe.br

O que é um paradigma?

Paradigma OO (Conceitos Básicos)

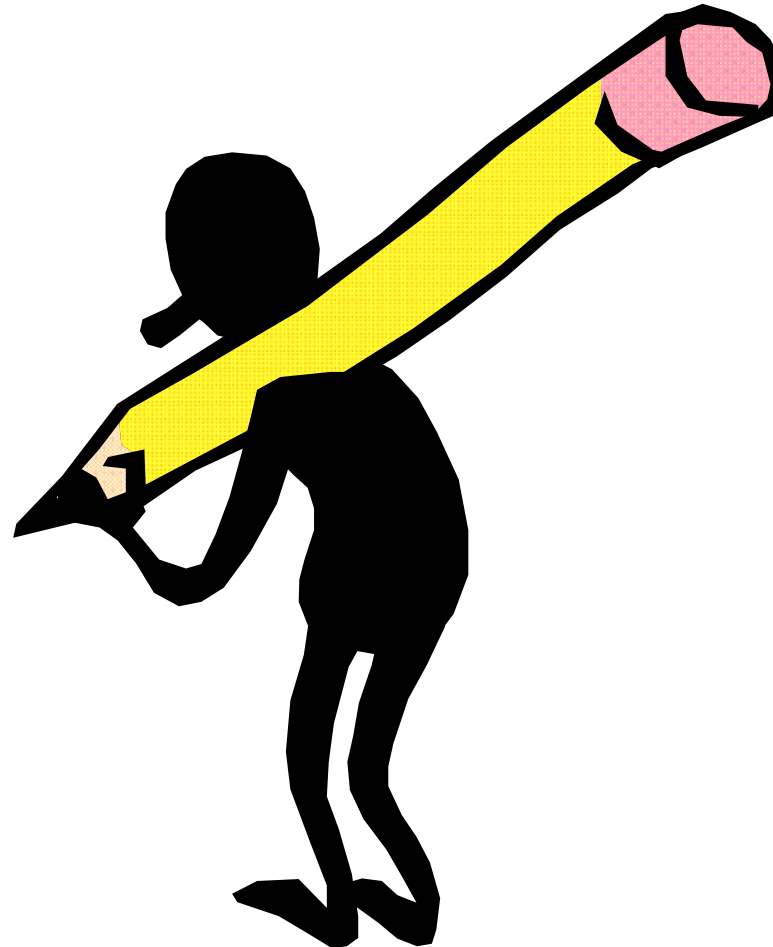
Paradigma

=

conjunto de teorias, métodos
e padrões que juntos representam
uma forma de organizar o
conhecimento, isto é, uma
forma de ver o mundo

Paradigma OO (Conceitos Básicos)

- Objeto
- Mensagem
- Encapsulamento
- Classe
- Herança
- Polimorfismo



Paradigma OO (Conceitos Básicos)

- Objeto
 - Alguma coisa (concreta ou abstrata) que faz sentido no domínio da aplicação e que desejamos manipular
 - É uma abstração (com significado definido) de uma coisa (tangível ou não) que existe no mundo real
 - Entidades distintas (concretas ou conceituais) sobre as quais armazenamos dados e temos operações para manipulá-los e são importantes dentro do contexto do problema a resolver
 - Exemplos:
 - O mouse X, o cliente Pedro, o Fusca de placa KKK-1234, a viagem Y, o voo Z, a disciplina laboratório de programação, a conta corrente 1111, a agência 2222, . . .

POO (Conceitos Básicos)

- Objeto
 - Um objeto tem um **ESTADO**, exibe um **COMPORTAMENTO** bem definido e tem uma **IDENTIDADE** única.

Mouse X



Tipo: ótico
Cor: preto e prata,
Botões: 2,
Scroll: sim
...

Estado



Clicar 1X,
Clicar 2X,
Arrastar,
...



Mouse X

Comportamento



Mouse B
Cada mouse é único!

Identidade

POO (Conceitos Básicos)

- Objeto (Estado de um objeto)
 - Conjunto de atributos que descrevem um objeto e seus relacionamentos
 - Conjunto das propriedades de um objeto associadas a seus valores correntes

Mouse X



Atributos/Propriedades

Tipo: ótico

Cor: preto e prata,

Botões: 2,

Scroll: sim

...

Estado

POO (Conceitos Básicos)

- Objeto (Comportamento de um objeto)
 - Conjunto de operações realizadas por um objeto
 - Conjunto de serviços que outros objetos podem requisitar
 - Operações são usadas para alterar o estado do próprio objeto

Operações

Clicar 1X,
Clicar 2X,
Arrastar,

...

Comportamento



Mouse X

POO (Conceitos Básicos)

- Objeto (Identidade de um objeto)
 - É uma referência única que distingue cada objeto (mesmo que este tenha os mesmos valores de atributos de outros objetos)
 - Propriedade de um objeto que o distingue de outros objetos
 - Esta é oculta aos usuários/programadores



Identidade

Existem muitos mouses iguais,
mas cada um é único!

POO (Conceitos Básicos)

- Objeto (Identidade de um objeto)
 - Igualdade de identidade (=) → para OIDs
 - Dois objetos possuem o mesmo OID
 - Igualdade de valor (= =) → para atributos
 - Dois objetos possuem o mesmo estado e comportamento, mas OID diferentes
 - Note que os mouses A, B, C, D e X são iguais, mas nenhum é idêntico a X!



X é igual a A,B,C e D? → V

X é idêntico a A,B,C e D? → F

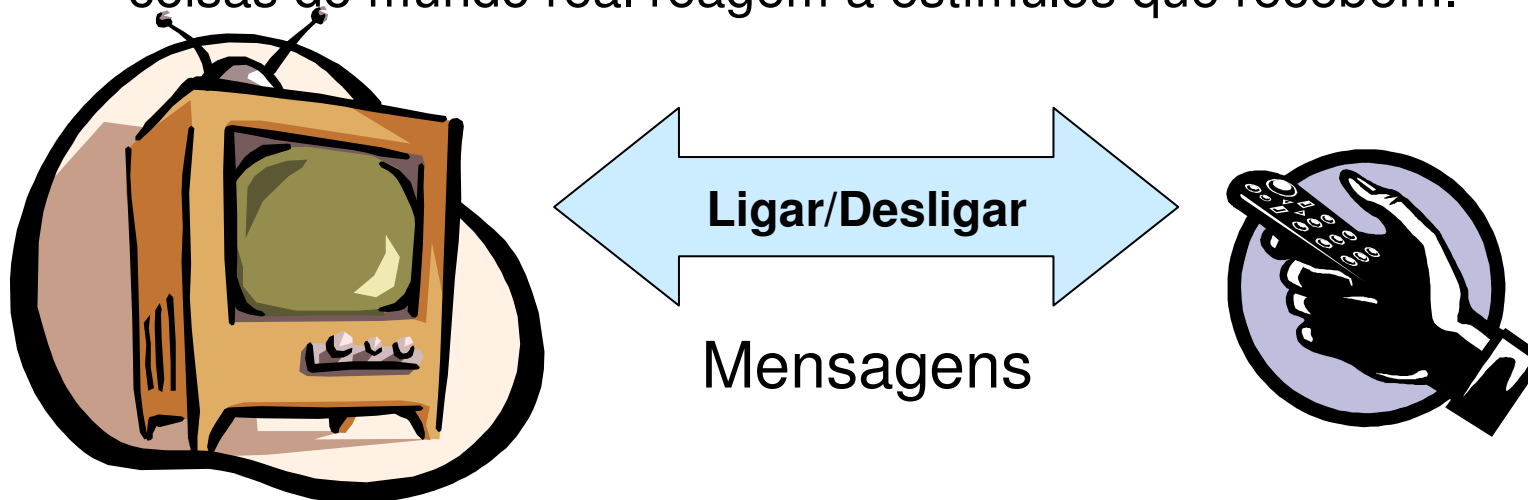
X é idêntico a X? → V

POO (Conceitos Básicos)

- Objeto (Identidade de um objeto)
 - Não se deve explicitar identificadores de objetos, pois estes não têm significado no mundo real
 - Não confunda identificadores internos com atributos identificadores do mundo real
 - O CPF, o número da placa do carro, o número do telefone, o RG não são identificadores internos. Eles são atributos identificadores do mundo real, pois têm significado no mundo real.

POO (Conceitos Básicos)

- Mensagem
 - É uma requisição enviada de um objeto (emissor) a outro (receptor) para que este último (receptor) realize alguma operação
 - Um objeto responde a estímulos a ele enviados, assim como as coisas do mundo real reagem a estímulos que recebem.



- Mensagem

- Objetos só podem se comunicar trocando mensagens!
 - Objetos trocam mensagens para realizar alguma tarefa dentro do sistema no qual eles estão inseridos
- Enviar uma mensagem é chamar uma operação de um objeto
 - A execução de uma operação é a interpretação que o objeto dá à mensagem
- Qual a diferença entre operação e mensagem?
 - Operação é a definição do que deverá ser realizado pelo objeto
 - Mensagem é a chamada ao serviço
- Exemplo:
 - Eu (objeto) sei programar... Se você (outro objeto) me pede para fazer um sistema, você estará me enviando uma mensagem

- Mensagens

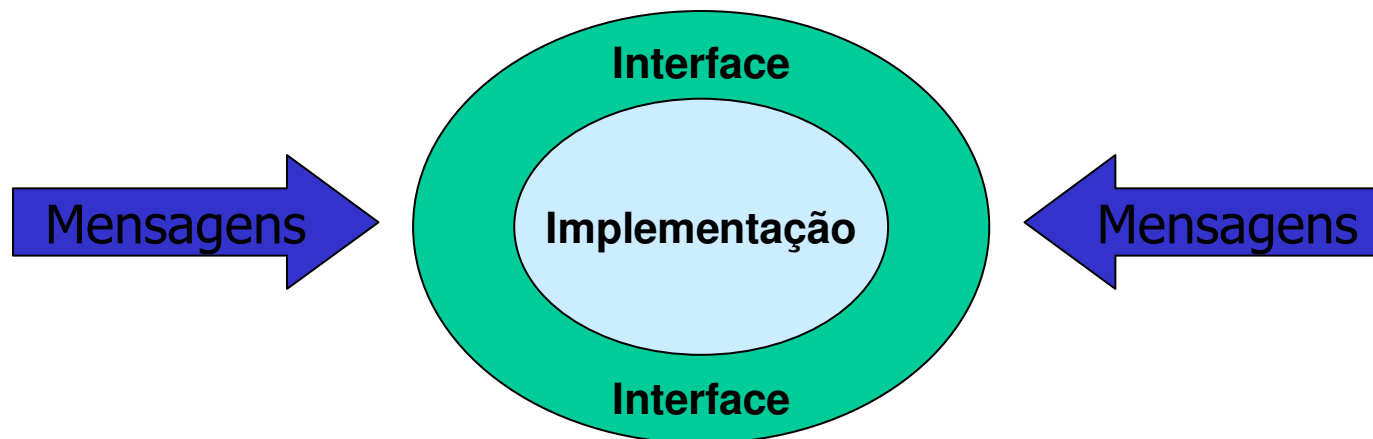
- O conjunto de mensagens que um objeto pode responder determina o seu comportamento.
 - Ao receber uma mensagem, o objeto executa a operação solicitada
- Notação genérica:
 - `receptor.operação(parâmetros);`
 - ↓ **É opcional**
 - Objeto** **Assinatura da operação**
- Exemplos:
 - `tv.ligar(); fusca.acelerar(80); pedro.InformarEstadoCivil();`

POO (Conceitos Básicos)

- Encapsulamento
 - Todo objeto tem uma parte que deve ser privada (atributos e operações) e uma parte que deve ser pública (interface).
 - A interface de um objeto é o que ele conhece e o que ele sabe fazer, sem descrever como o objeto conhece ou faz
 - A interface de um objeto define as operações que ele pode realizar e conseqüentemente as mensagens que ele recebe

POO (Conceitos Básicos)

- Encapsulamento
 - É através da interface de um objeto receptor que um objeto requisitante sabe que mensagens podem ser enviadas



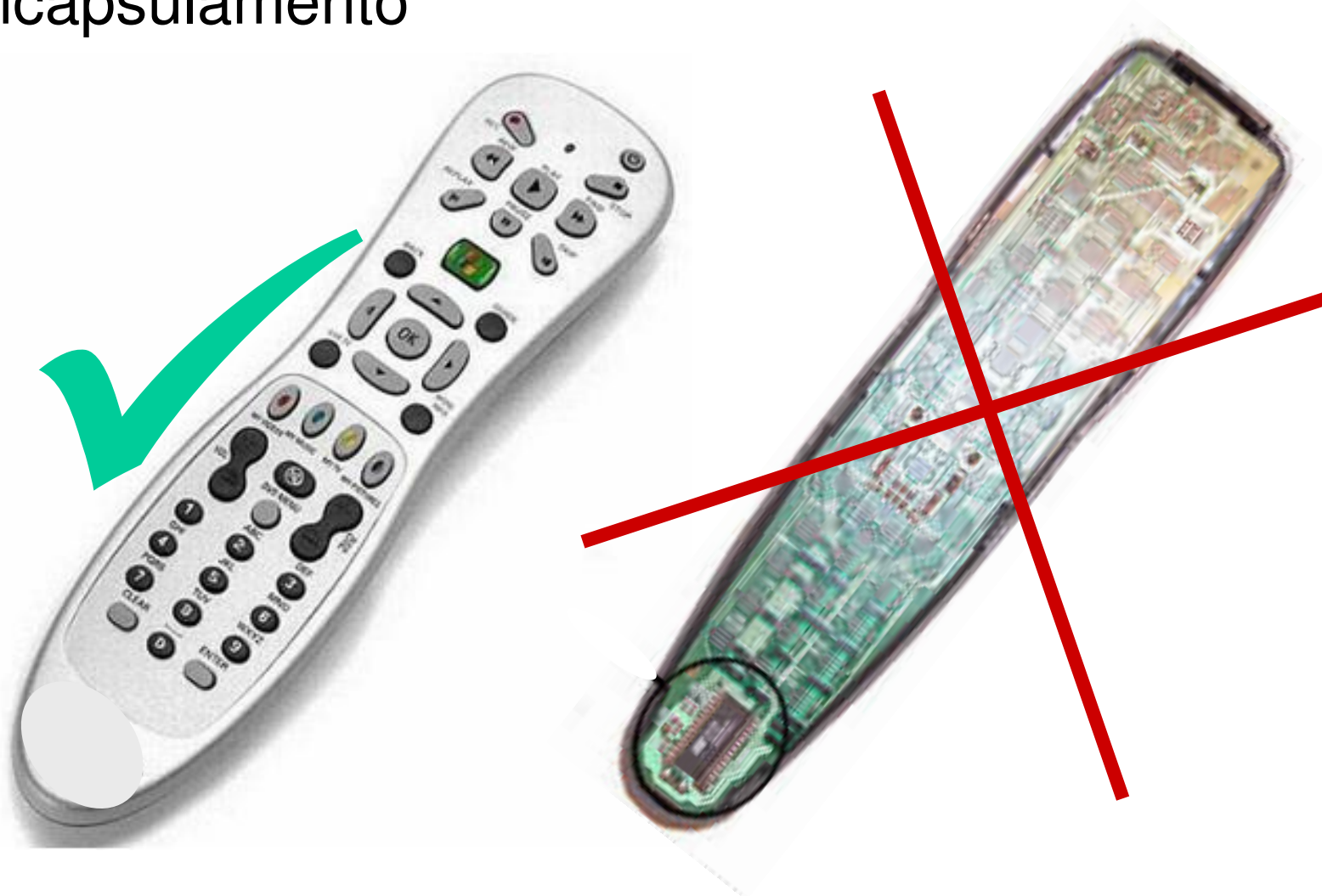
POO (Conceitos Básicos)

- Encapsulamento
 - Consiste em “esconder” os detalhes de implementação do objeto, permitindo que mudanças na implementação das operações não afetem a interface dessas operações
 - É uma forma de restringir o acesso ao comportamento interno de um objeto (oculta os detalhes internos de implementação do objeto)
 - Encapsulamento permite ocultação de informação

- Encapsulamento
 - Um objeto que precise se comunicar como outro objeto para realizar alguma tarefa não precisa saber como esta tarefa é implementada pela sua parte privada
 - Utilizamos um objeto sem necessitar conhecer seu funcionamento interno
 - Para interagir com um controle de TV não precisa-se conhecer seu funcionamento interno. Basta apenas saber o que é permitido fazer com o ele e fazer
 - De forma semelhante faz-se com um objeto. Isto é, basta conhecer sua interface para interagir com o mesmo

POO (Conceitos Básicos)

- Encapsulamento



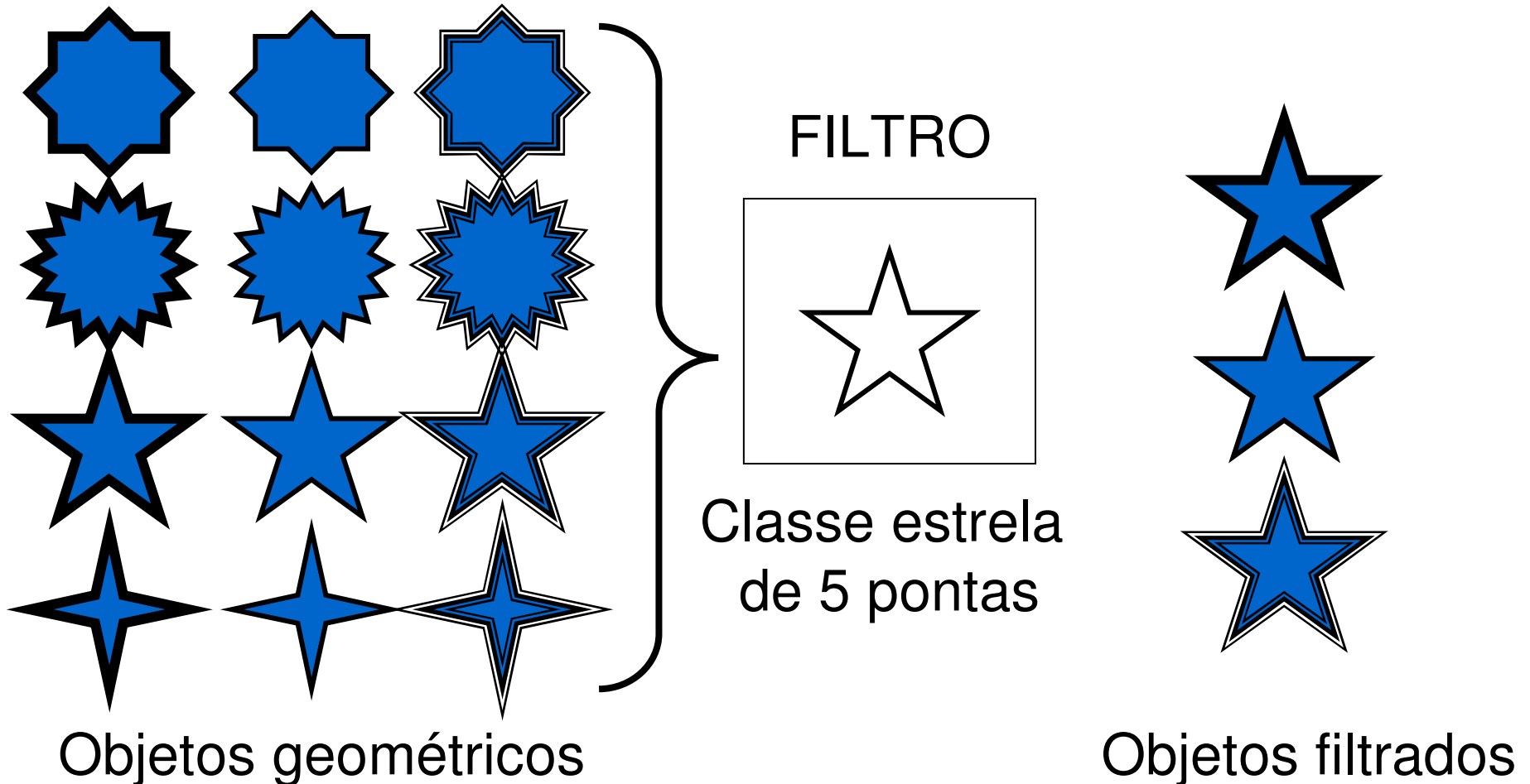
- Classe
 - É a representação de um agrupamento de objetos que compartilham o mesmo estado e comportamento
 - Seres humanos agrupam os objetos para entendê-los
 - Objetos similares são agrupados em classes
 - É uma abstração das características mais importantes de um grupo de objetos semelhantes

- Classe - Exemplos

- Objetos físicos ou tangíveis
 - Aeronave, Carro, Livro, DVD
- Lugares
 - Aeroporto, Loja, Empresa, Livraria, Banco
- Transações
 - Venda, Pagamento, Reserva, Pedido
- Itens da Transação
 - ItemVenda, ItemLocacao, ItemPedido, ItemPagamento
- Papéis desempenhados pelas pessoas
 - Funcionário, Fornecedor, Cliente, Caixa, Professor, Aluno
- Organizações e Órgãos
 - Departamento, Linha Aérea, Sala, Laboratório
- Eventos
 - Roubo, Reuniao, Acidente

POO (Conceitos Básicos)

- Classes e Objeto
 - Uma classe pode ser entendida como um filtro de objetos



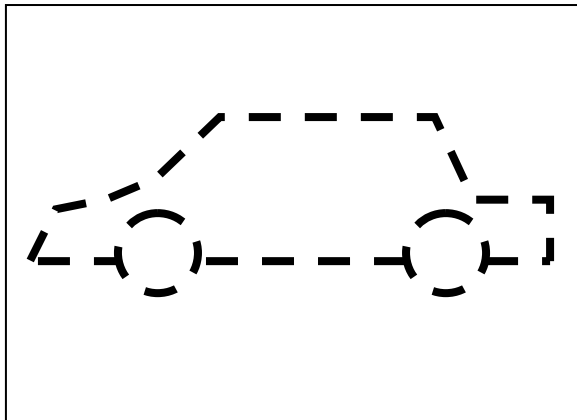
POO (Conceitos Básicos)

- Classes e Objeto
 - Uma classe pode ser entendida como um molde de objetos
 - Uma classe pode ter mais de um molde (operação construtora)
 - A criação de um objeto é feita a partir de um operação construtora

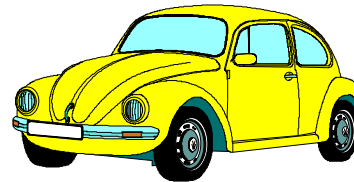


POO (Conceitos Básicos)

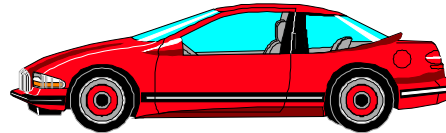
- Classes e Objeto
 - Diz-se que um objeto é uma instância de uma classe.
 - Todas as instâncias de uma classe têm valores próprios para os atributos especificados na classe



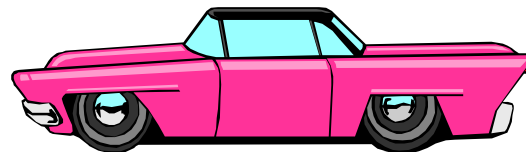
Classe Carro



Objeto Fusca



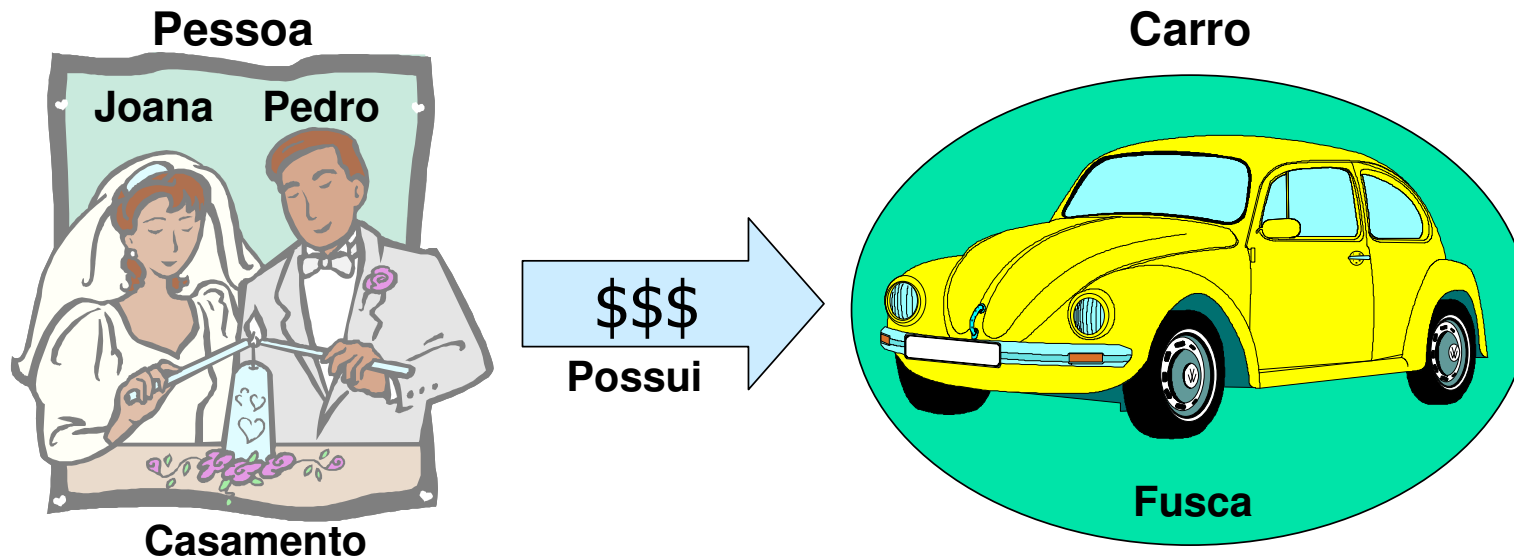
Objeto Ferrari



Objeto Cadilque

POO (Conceitos Básicos)

- Classe e Objeto
 - Classe = mecanismo de Abstração “Classificação”
 - Objeto = mecanismo de Abstração “Instanciação”
 - Objetos de uma Classe podem se relacionar entre si e com Objetos de outras Classes
 - Joana e Pedro são pessoas decentes. Tempos atrás, Joana casou-se com Pedro. Pedro possui um carro fusca amarelo 1976



- Classes e Objeto
 - Cada objeto conhece sua classe. A maioria das LPOO permite determinar a classe de um objeto em tempo de execução (Ligação tardia ou Late Binding)
 - A classe de um objeto é uma propriedade implícita do objeto
 - As definições comuns (como o nome da classe, os atributos, os relacionamentos e as operações) são definidas uma vez por classe, ao invés de uma vez por instância (objeto)
 - Isto permite que os objetos da classe se beneficiem da reutilização do código escrito

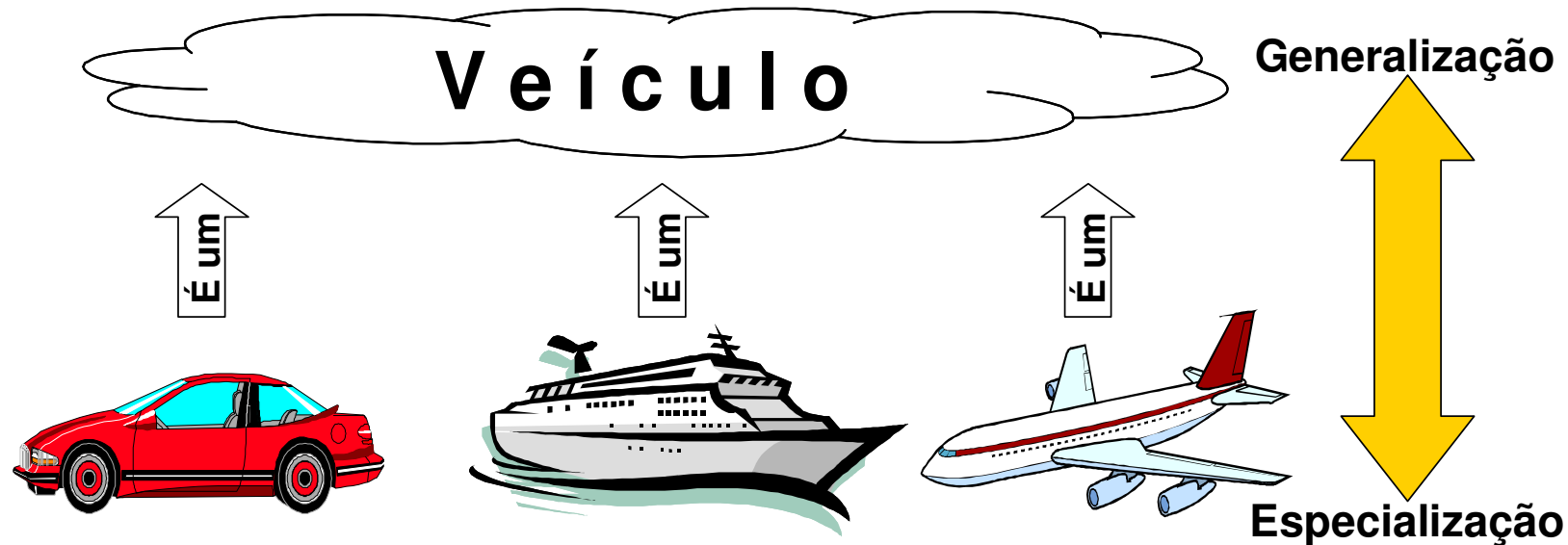
POO (Conceitos Básicos)

- Classes e Objeto
 - Classe é uma definição estática das principais características (estado + comportamento) de um grupo de objetos semelhantes
 - Objeto é algo dinâmico (“tem vida”) e obedece as definições de sua classe
 - Um objeto é algo que pode ser instanciado, manipulado e destruído
 - Ex:
 - classe: cadeira;
 - objeto: a que você está sentado

- Herança
 - Mecanismo que permite definir uma nova classe (subclasse) a partir de uma classe (superclasse) já existente
 - A subclasse deve estender a superclasse
 - A subclasse herda as características comuns da superclasse
 - A subclasse pode reescrever métodos herdados da superclasse
 - É uma relação transitiva
 - Se A herda de B e C herda de B, então C também herda de A
 - Classes semelhantes são organizadas em hierarquias

POO (Conceitos Básicos)

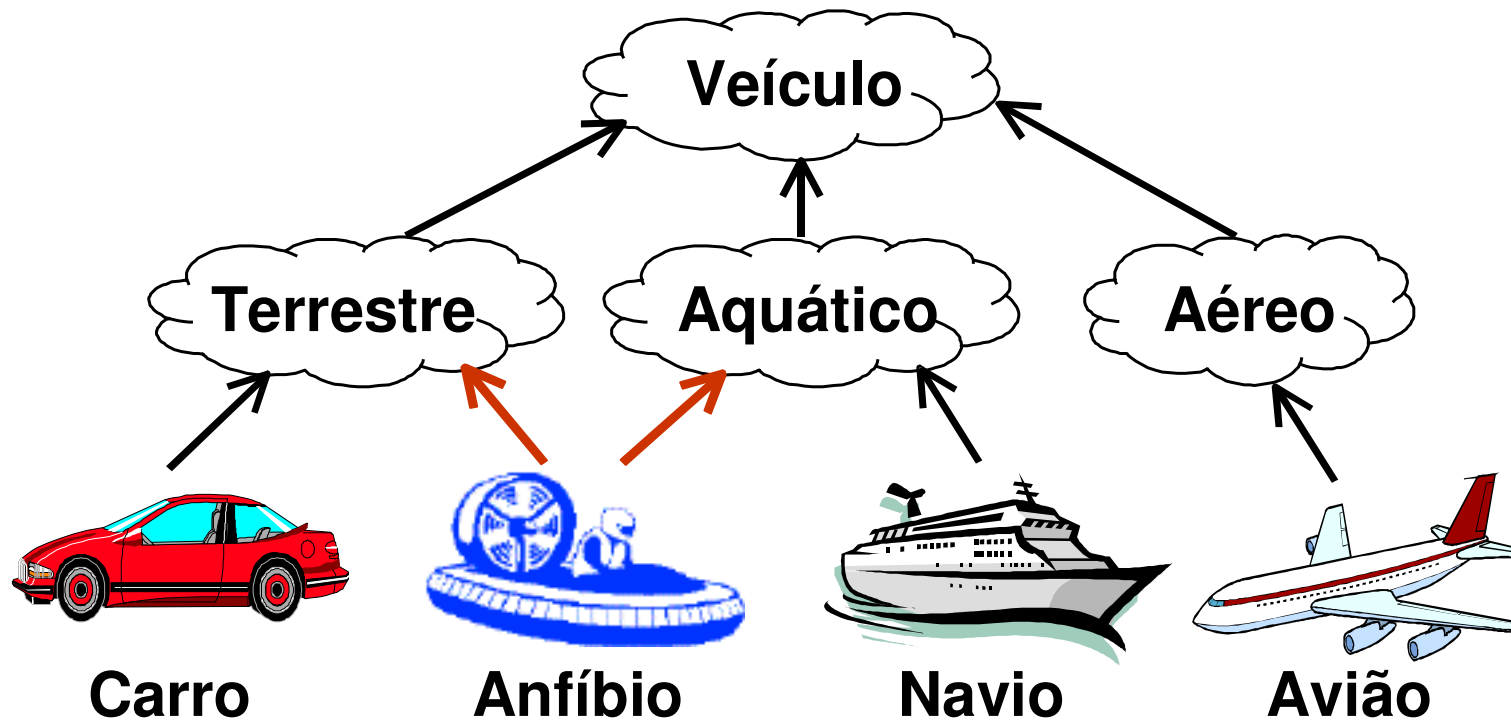
- Herança
 - Baseada no mecanismo de Abstração
Generalização/Especialização



- A herança pode ser vista como um nível de abstração, onde cada nível de uma hierarquia pode ser visto como um nível de abstração

POO (Conceitos Básicos)

- Herança
 - Pode ser de dois tipos
 - Simples: uma classe é subclasse de somente uma superclasse
 - Múltipla: uma classe é subclasse de mais de uma superclasse

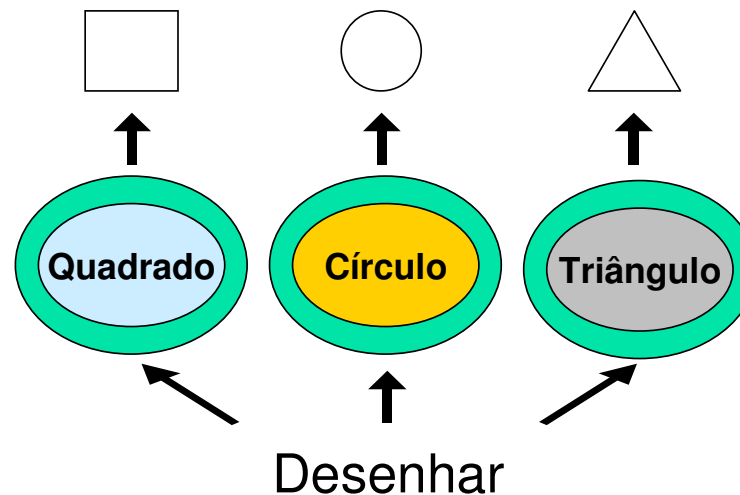


- Polimorfismo
 - Uma mensagem diz qual operação de um objeto deve ser realizada
 - Como a responsabilidade de executar a mensagem é do Receptor, pode acontecer que uma mesma mensagem ativar operações diferentes
 - Por exemplo, a mensagem desenhar será realizada de forma diferente pelas classes quadrado, círculo e triângulo

POO (Conceitos Básicos)

- Polimorfismo

- É a habilidade de objetos responderem a uma mesma mensagem de diferentes maneiras
- Permite a criação de classes com interfaces idênticas, porém seus objetos têm implementações de operações que são diferentes



POO (Conceitos Básicos)

- Polimorfismo (sobrecarga X sobre-escrita)
 - O polimorfismo pode ser obtido a partir de sobrecarga ou sobre-escrita de operações
 - Sobrecarga de operações
 - O polimorfismo ocorre dentro de uma classe
 - As operações têm o mesmo nome, mas assinaturas diferentes
 - Permite diferentes implementações, de operações similares, dentro da mesma classe
 - Sobre-escrita de operações
 - O polimorfismo ocorre entre uma super-classe e suas sub-classes
 - As operações têm o mesmo nome e as mesmas assinaturas, mas implementações diferentes
 - Permite diferentes implementações, de operações similares, entre classe diferentes

POO (Conceitos Básicos)

- Polimorfismo (Late Binding X Early Binding)
 - O polimorfismo só é possível graças ao suporte da ligação tardia (late binding) de chamadas de operações
 - Em tempo de compilação (Ligação Prematura ou Early Binding), não tem como haver polimorfismo!

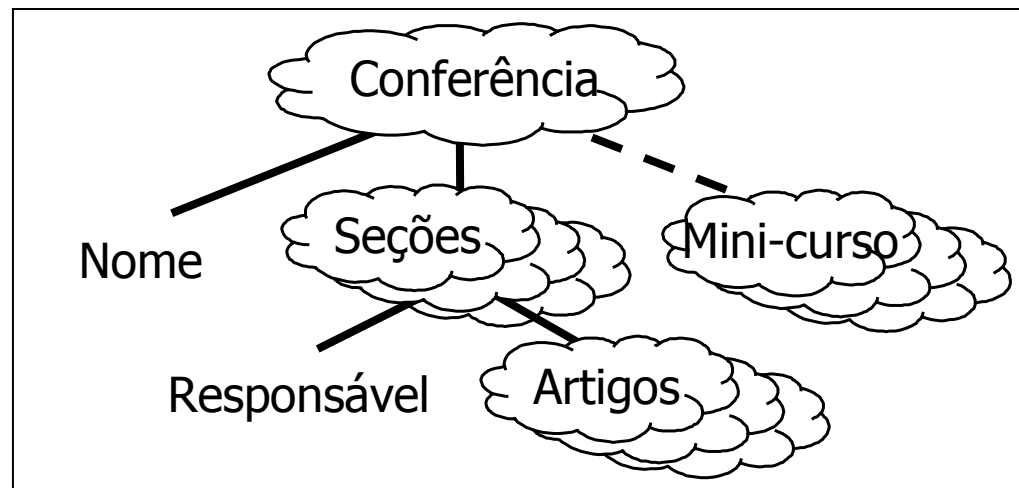
- Os objetos em um BD podem ser:
 - Persistentes
 - São armazenados após a execução de um programa
 - Podem ser recuperados e compartilhados com outros programas
 - Transientes
 - Não são armazenados, pois só existem somente durante a execução de um programa

- Os OID em um BD:
 - Realizam a correspondência direta entre objetos do mundo real e objetos do BD
 - Objetos têm seu estado atualizado sem perder seu OID
 - É independente do endereço de armazenamento
 - É desejável que seja utilizado apenas uma vez (mesmo após a exclusão do objeto)
 - Possibilita compartilhamento

- Os OID em um BD:
 - OID não é chave (mas em OR uma chave pode ser OID)
 - Uma chave natural tem semântica!
 - O conceito de chave deve ser preservado!
 - Uma chave é definida pelo valor de um ou mais atributos e pode sofrer modificações
 - Chaves não são únicas no sistemas, são únicas em uma relação
 - OID são únicos no sistema

- Os OID em um BD :
 - Em SGBDR, quando a PK é a combinação de vários campos, os usuários preferem introduzir algum identificador artificial para usá-lo como FK
 - Nestas situações, OIDs de SGBDOO/OR são mais interessantes (facilitam a manipulação)

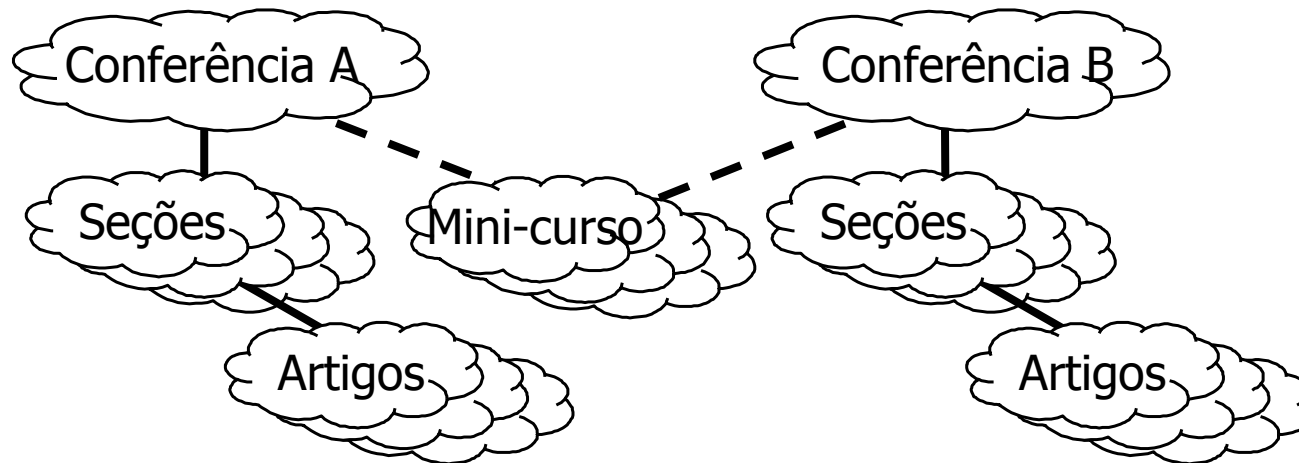
- Objetos Complexos
 - Objetos podem ser agregados para formar outros objetos
 - EX.: Uma conferência tem várias seções e mini-cursos
 - Agrupamento pode ocorrer em diversos níveis
 - EX.: Artigos têm autores e são apresentados em seções



- Objetos Complexos:
 - Semântica de Propriedade
 - Os objetos de um objeto complexo são parte dele
 - Relacionamento: é-parte (agregação) ou é-componente-de (composição)
 - Semântica de Referência
 - Os objetos relacionados a um objeto complexo são independentes dele
 - Relacionamento: é-associado-a (associação)

Detalhes do POO em BD

- Objetos Complexos:
 - Semântica de Propriedade X Semântica de Referência



Legenda:

Semântica de Propriedade ———

Semântica de Referência - - - -

- Objetos Complexos:
 - Estruturados
 - Obtidos com os construtores de objetos
 - SGBD conhece a estrutura do objeto
 - Não-Estruturados
 - Objetos não interpretados Ex.: imagens, som, etc
 - SGBD não conhece a estrutura do objeto

- Objetos Complexos:
 - Vantagens:
 - Representação direta dos objetos como textos, mapas, desenhos
 - Sem necessidade de decomposição em unidades menores (tuplas)
 - Recuperação, atualização, armazenamento e exclusão de dados e mais fácil e mais direta
 - Contudo, ter atenção com:
 - A granularidade dos objetos e
 - O acoplamento entre os objetos

cin.ufpe.br



Centro de Informática

U • F • P • E



UNIVERSIDADE FEDERAL DE PERNAMBUCO