

Desenvolvimento Orientado a Objetos

Prof.: Edson dos Santos Cordeiro

Prof. Edson S. Cordeiro

Bibliografia

- ▶ LARMAN, Graig. Utilizando UML e padrões. Porto Alegre: Bookman, 2000.
- ▶ STAA, Arndt von. Programação modular. Rio de Janeiro: Campus, 2000.
- ▶ BORATTI, Isaias Camilo. Programação Orientada a Objetos. Florianópolis: Visual Books 2001.
- ▶ SCHMITZ, Eber; SILVEIRA, Denis. Desenvolvimento de Software Orientado a Objetos. Rio de Janeiro: Brasport, 2000.

11/08/2002 Prof. Edson S. Cordeiro 2

Bibliografia

- ▶ DEITEL, H. M.; DEITEL, P. J. Java: como programar. Porto Alegre: Bookman, 2001.
- ▶ McCARTY, Bill; GILBERT, Stephen. Object-Oriented desing in Java. Corte Madera, CA: Mitchell Waite, 1999.
- ▶ SEBESTA, R. W. Conceitos de linguagens de programação. Bookman, 2000.
- ▶ JAMSA, K.; KLANDER, L. Programando em c/c++: a biblia. Makron Books,1999.

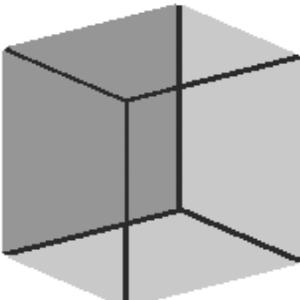
11/08/2002 Prof. Edson S. Cordeiro 3

Introdução

- ▶ Primeiras linguagens surgiram em 50;
- ▶ Simula no início de 60 empregou conceitos orientado a objetos;
- ▶ Início de 70 surgiu o conceito de Tipos de Dados Abstratos (TDA);
- ▶ Entre os anos 70 e 80, conceitos orientados a objetos foram consolidados (SmallTalk).

11/08/2002 Prof. Edson S. Cordeiro 4

O que você vê ?



O azul está na parte de dentro, à esquerda
ou
na parte de fora, à frente?

- ⇒ Katimba era um excelente caçador.
- ⇒ Katimba ficou conhecido na aldeia como um excelente projetista de lanças.



⇒ Depois de algum tempo.

- ⇒ Katimba somente projetava lanças para atender sua aldeia.
- ⇒ Katimba não tinha mais tempo para caçar.



- ⇒ Mais um tempo se passou.
- ⇒ Katimba continua projetando lanças.




 Será que Katimba ainda é um bom projetista de lanças?



Modelagem Orientada a Objetos

- ▶ Uniformização dos formalismos utilizados na análise e projeto.
- ▶ Notação unificada (UML): É utilizado uma mesma notação entre as diversas fases do projeto de desenvolvimento

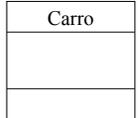
Gap Semântico

- ▶ Diferença entre Espaço de Problemas e Soluções.

Espaço de problema



Espaço de solução



Diminuir Gap Semântico

Abordagem Orientada a Objetos

- ▶ Tradicionalmente
 - ▶ Softwares são formados por dados e procedimentos independentes.
- ▶ Paradigma OO
 - ▶ Softwares são formados, principalmente, por:
 - ▶ Objetos
 - ◆ parte interna (define o comportamento do objeto, ou seja, especifica os métodos associados ao objeto)
 - ◆ parte externa (interface - maneira que o objeto se apresenta para outros objetos)
 - ▶ Mensagens.

11/08/2002

Prof. Edson S. Cordeiro

13

Modelo Conceitual

- ▶ Tarefa mais importante de um processo de desenvolvimento de software:
 - ▶ Análise e Modelagem do domínio da aplicação independentemente da implementação.
- ▶ A tarefa de modelagem conceitual envolve dois mecanismos:
 - ▶ Abstração e,
 - ▶ Representação.

11/08/2002

Prof. Edson S. Cordeiro

14

Abstrações e Representações

Casa



11/08/2002

Prof. Edson S. Cordeiro

15

Processo de Abstração



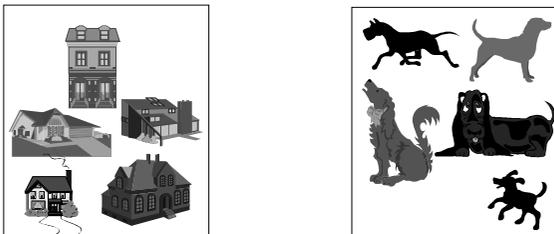
11/08/2002

Prof. Edson S. Cordeiro

16

Processo de Abstração

Classificação: categorização dos objetos em grupos e/ou classes (categoria), com base em algum conjunto de propriedades comuns.



11/08/2002

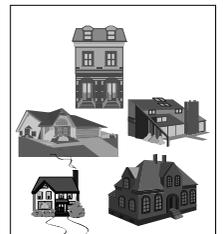
Prof. Edson S. Cordeiro

17

Processo de Abstração

Generalização (Especialização)

- A partir de dois ou mais objetos, abstrai-se uma categoria mais genérica.
- sub-categorias satisfazem todas as propriedades das categorias de que elas constituem especializações.
- deve existir pelo menos uma propriedade que distinga duas categorias especializadas.



11/08/2002

Prof. Edson S. Cordeiro

18

Processo de Abstração

Generalização

Telhado

Portas

Janelas

Especialização

Casa de campo

Residencial

Casa de praia

11/08/2002 Prof. Edson S. Cordeiro 19

Processo de Abstração

Agregação / Decomposição

- Composição de uma nova categoria como um agregado de categorias pré-existentes.
- Decomposição de uma categoria agregada em outras categorias atômicas.

11/08/2002 Prof. Edson S. Cordeiro 20

O que são Objetos ?

- ▶ Um objeto pode ser considerado uma “coisa” que executa um conjunto de atividades;
- ▶ Um conjunto de atividades que pode ser executada pelo objeto define o comportamento desse objeto;
- ▶ Por exemplo: uma calculadora executa um conjunto de atividades (somar, subtrair, multiplicar, dividir ...). Esse conjunto de atividades define como a calculadora pode se comportar.

11/08/2002 Prof. Edson S. Cordeiro 21

O que são Objetos ?

- ▶ Um objeto é como uma “caixa preta” que pode receber e/ou enviar mensagens;
- ▶ Uma caixa preta pode conter código (uma seqüência de instruções de computador) e dados (informações que são manipuladas pelas instruções);
- ▶ Note que instruções e dados são mesclados em uma única coisa indivisível: um objeto.

11/08/2002 Prof. Edson S. Cordeiro 22

O que são Objetos ?

- ▶ A interface do objeto consiste em um conjunto de comandos;
- ▶ Cada comando executa uma ação específica;
- ▶ Um objeto pode solicitar a outro objeto que execute uma ação por meio do envio de uma mensagem;

11/08/2002 Prof. Edson S. Cordeiro 23

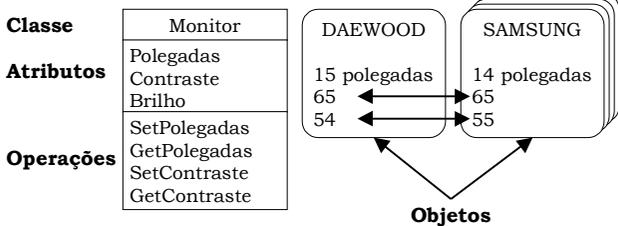
O que são Objetos ?

- ▶ Como o objeto receptor interpreta as mensagens enviadas por outro objeto?
- ▶ Como as mensagens são processadas pelo objeto?

11/08/2002 Prof. Edson S. Cordeiro 24

O que são Atributos ?

- ▶ Todas as instâncias de uma classe têm valores próprios para os atributos especificados na classe;
- ▶ Os objetos representados por determinada classe diferenciam-se entre si pelos valores de seus atributos.



11/08/2002

Prof. Edson S. Cordeiro

25

O que são Atributos ?

- ▶ Representam um conjunto de informações;
- ▶ Cada objeto possui seu próprio conjunto de atributos;
- ▶ São variáveis que definem o estado de um objeto, ou seja, caracterizam os objetos (situação do celular);
- ▶ Descrevem as informações que ficam escondidas em um objeto para serem exclusivamente manipuladas pelas operações daquele objeto (Visibilidade - Encapsulamento);

11/08/2002

Prof. Edson S. Cordeiro

26

O que são métodos ?

- ▶ Cada mensagem tem um código (instruções) associada a ela;
- ▶ Quando um objeto recebe uma mensagem, um código (instruções) é executado;
- ▶ Em outras palavras, essas mensagens determinam o comportamento do objeto e o código determina como o objeto irá responder a cada uma das mensagens recebidas.

11/08/2002

Prof. Edson S. Cordeiro

27

O que é Mensagem ?

- ▶ Mecanismo de comunicação entre objetos através do qual se desencadeia a execução de uma tarefa.
- ▶ As mensagens existem somente em função dos serviços, e fazem o mapeamento:
 - ▶▶ de um objeto para outro objeto;
 - ▶▶ de um objeto para uma classe (criação de objetos) e,
 - ▶▶ de uma classe para outra classe (criação de objetos dentro de outros objetos).

11/08/2002

Prof. Edson S. Cordeiro

28

O que são métodos ?

- ▶ O código associado a cada mensagem é chamado de método;
- ▶ Quando um objeto recebe uma mensagem, ele determina qual método irá responder e passa o controle para esse método;
- ▶ Métodos são similares às subrotinas, procedimentos ou funções encontradas em linguagens procedurais como C e Pascal.

11/08/2002

Prof. Edson S. Cordeiro

29

O que são métodos ?

- ▶ Por exemplo: o nome de um método é similar ao nome de uma subrotina e o código do método é similar ao código encontrado em uma subrotina;
- ▶ Enviar uma mensagem para um objeto é similar a chamar uma subrotina.

11/08/2002

Prof. Edson S. Cordeiro

30



O que são Objetos ? Parte II

- ▶ Cada objeto precisa manter as informações de como executar um determinado comportamento;
- ▶ Apenas os métodos de um objeto podem manipular os valores armazenados em suas variáveis.



O que é encapsulamento ?



- ▶ Problema:
 - ▶ O que fazer quando o programa atinge milhares de linhas de código ? Programas extensos ...
 - ▶... dificultam a organização do código;
 - ▶... aumentam custo na compilação do programa.



- ▶ Solução (entre outras):
 - ▶ Modularização: organizar o programa em containers que incluem subprogramas e dados relacionados logicamente compiláveis separadamente.



O que é encapsulamento ?

- ▶ É um agrupamento de subprogramas e dos dados que eles manipulam (Sebesta, p. 397);
- ▶ O encapsulamento permite uma melhor organização lógica para uma coleção de computações relacionadas;



O que é encapsulamento ?

- ▶ Um objeto não pode manipular as variáveis de outros objeto diretamente;
- ▶ Um objeto apenas pode acessar variáveis de outro objeto mediante ao envio de mensagens para esse objeto;
- ▶ Esse recurso é chamado de encapsulamento e garante um processo seguro para a manipulação das variáveis de um objeto.



O que é encapsulamento ?

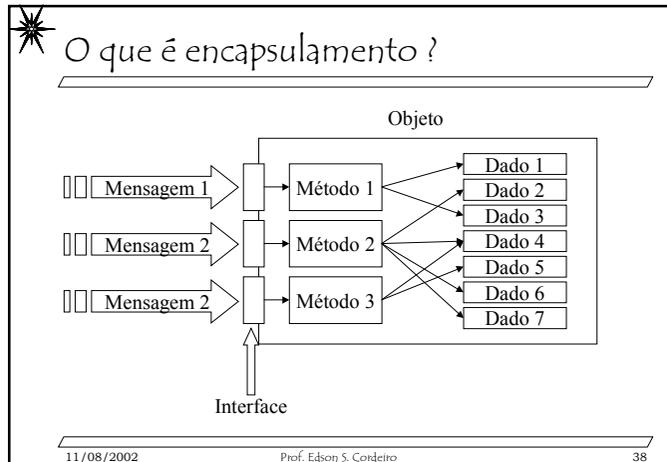
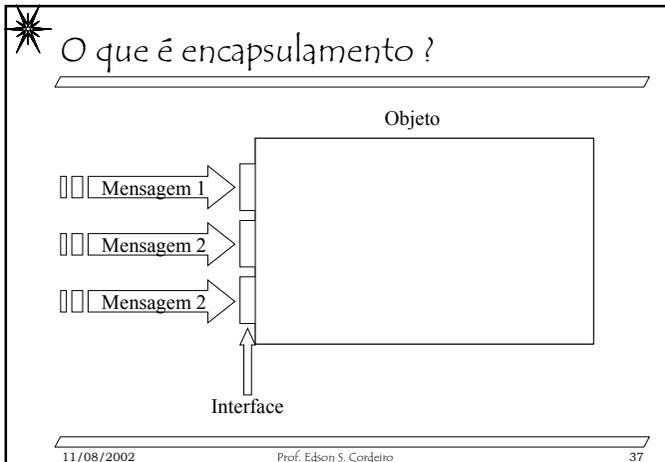
- ▶ Diferentemente das linguagens procedurais onde áreas comuns de dados são utilizadas para compartilhamento de informações, a programação orientada a objetos não recomenda o acesso direto à áreas comuns de dados;
- ▶ Apenas o objeto proprietário do dado (variável) pode manipular seu conteúdo. Outros objetos podem manipular os dados somente por meio das mensagens enviadas ao objeto proprietário do dado (variável).



O que é encapsulamento ?

- ▶ Na Programação Orientada a Objetos (POO), objetos interagem com outros objetos apenas por meio de mensagens;
- ▶ A única coisa que um objeto conhece sobre outro objeto é a interface do objeto;
- ▶ A lógica e dados de um objeto são escondidos de outros objetos;
- ▶ A interface encapsula os dados e código de um objeto.





- ### O que é encapsulamento ?
- ▶ O encapsulamento permite ao desenvolvedor separar a implementação do objeto de seu comportamento;
 - ▶ A separação o efeito “caixa-preta” onde o usuário do objeto é isolado mudanças decorridas de implementações;
 - ▶ Qualquer mudança interna ao objeto não transparece ao usuário do objeto.
- 11/08/2002 Prof. Edson S. Cordeiro 39

- ### O que é encapsulamento ?
- ▶ Objetos encapsulam suas estruturas de dados (atributos).
 - ▶ Restringe a visibilidade do objeto.
 - ▶ Torna inacessível os detalhes de manipulação do objeto de seu meio externo.
 - ▶ Os usuários do objeto compreendem quais operações podem ser solicitadas de um objeto, mas não conhecem os detalhes de como a operação é executada.
- 11/08/2002 Prof. Edson S. Cordeiro 40

- ### O que é Visibilidade ?
- ▶ Cada um dos elementos que constituem uma classe possui uma visibilidade (Staa, p. 296);
 - ▶ Visibilidade é a capacidade de um objeto poder “ver” ou de ter referência a outro objeto (Larman, p. 246);
 - ▶ O conceito de visibilidade também é conhecido como modificadores de acesso.
- 11/08/2002 Prof. Edson S. Cordeiro 41

- ### O que é Visibilidade ?
- ▶ Os tipos de visibilidade que podem ser determinados em uma classe para atributos e operações são:
 - + **public**: os elementos são acessíveis por todas as classes;
 - # **protected**: os elementos são acessíveis por subclasses, ou pela própria classe;
 - **private**: os elementos são acessíveis somente pela própria classe;
- 11/08/2002 Prof. Edson S. Cordeiro 42

O que é Visibilidade ?

- ▶ **Atributos:**
 - ▶ Atributos somente deveriam ser acessados pelos métodos da classe a que pertence;
- ▶ **Operações:**
 - ▶ Públicas (interface);
 - ▶ Privadas (internas à classe).

Classe

Operações

Dados

11/08/2002 Prof. Edson S. Cordeiro 43

O que é polimorfismo ?

- ▶ Outro benefício decorrente da separação da implementação do comportamento é o polimorfismo;
- ▶ O polimorfismo permite que dois ou mais objetos respondam a mesma mensagem diferentemente.

11/08/2002 Prof. Edson S. Cordeiro 44

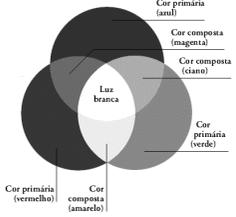
O que é polimorfismo ?



11/08/2002 Prof. Edson S. Cordeiro 45

O que é Herança ?

- ▶ **Problema:**
 - ▶ Como aumentar a produtividade ?
- ▶ **Solução (entre outras):**
 - ▶ Reutilizar código (instruções) já existentes.



11/08/2002 Prof. Edson S. Cordeiro 46

O que é Herança ?

Como realizar pequenas modificações em I1 para reutilizar em B ?

Para construir B preciso reutilizar apenas A1?

Módulo A
Variável A1
Variável A2
Instrução I1

Módulo B
????



11/08/2002 Prof. Edson S. Cordeiro 47

O que é Herança ?

- ▶ Mecanismo que permite definir uma nova classe (subclasse) a partir de uma classe já existente (superclasse).
- ▶ Ao se estabelecer uma Especialização (subclasse) de uma classe, a subclasse herda as características comuns da superclasse, isto é, a especificação dos atributos e dos métodos da superclasse passam a fazer parte da especificação dos atributos e dos métodos da subclasse.
- ▶ A subclasse pode adicionar novos métodos e/ou atributos, como também reescrever (sobrepôr) métodos herdados.

11/08/2002 Prof. Edson S. Cordeiro 48

O que é Herança ?

- ▶ Quando uma mensagem é enviada para um objeto:
 - ▶ A procura do método correspondente começa pela classe do objeto;
 - ▶ Se o método não for encontrado, a procura continua na superclasse.
- ▶ A Herança pode ser de dois tipos:
 - ▶ Herança Simples: quando uma classe é subclasse de somente uma superclasse e,
 - ▶ Herança Múltipla: quando uma classe é subclasse de várias superclasses e conseqüentemente herda as características de cada uma delas

11/08/2002

Prof. Edson S. Cordeiro

49

O que é Herança ?

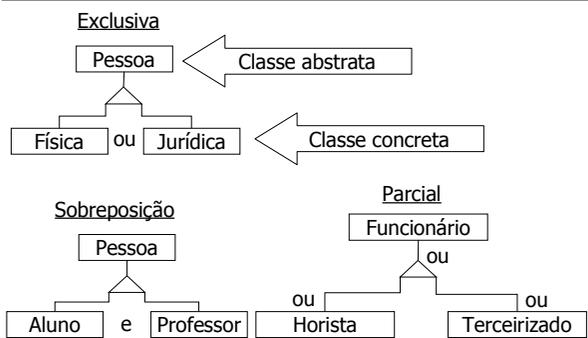
- ▶ Cobertura da herança:
 - ▶ Total/Exclusiva:
 - ▶ ocorre quando é instanciada apenas a subclasse do conjunto de subclasses da superclasse.
 - ▶ Parcial:
 - ▶ ocorre quando somente a superclasse é instanciada, não havendo necessidade de instanciar a subclasse.
 - ▶ Sobreposição:
 - ▶ ocorre quando mais de uma subclasse é instanciada.

11/08/2002

Prof. Edson S. Cordeiro

50

O que é Herança ?



11/08/2002

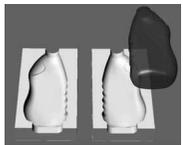
Prof. Edson S. Cordeiro

51

"Forjar" Classes



- ▶ Classes transcendem objetos;
- ▶ Resultado: "Moldes";
- ▶ As classes moldam características dos objetos;
- ▶ Objetos se comportam de acordo com o comportamento da classe "molde".



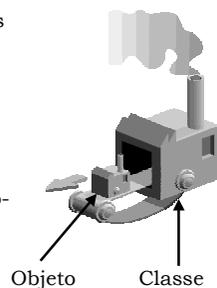
11/08/2002

Prof. Edson S. Cordeiro

52

Categorização em Classes

- ▶ Quando estamos modelando um domínio, observamos que há várias entidades similares que podem ser abstraídas em um conceito único que engloba esta similaridade;
- ▶ O POO incorporou esta operação através da abstração dos atributos e dos métodos que caracterizam objetos semelhantes, especificando-os na CLASSE.



11/08/2002

Prof. Edson S. Cordeiro

53

Operação

- ▶ **Operação (Método)**
- ▶ É aplicada ao objeto, ativada pelo envio de uma mensagem. São executadas operações sempre que um objeto recebe uma mensagem.
- ▶ As operações podem ser classificadas em categorias:
 - ▶ Construtoras;
 - ▶ Modificadoras;
 - ▶ Destrutoras.

11/08/2002

Prof. Edson S. Cordeiro

54



Métodos Construtores

- ▶ Métodos Construtores são utilizados para:
 - ▶ Atribuem valores padrões (default) para os atributos do objeto que será criado;
 - ▶ Criarem objetos que compõem uma classe.
- ▶ Podem ser implementados com ou sem argumentos de entrada;
- ▶ Métodos construtores não retornam valores;
- ▶ **Importante:** Objetos com atributos do tipo classe devem chamar (mensagem) o método construtor do atributo do tipo classe.



Métodos Destrutores

- ▶ Tem por objetivo destruir o objeto, ou seja, liberar o espaço de memória alocado para esse objeto.
- ▶ Após a destruição, o objeto não poderá receber nenhuma mensagem.
- ▶ **Importante:** Objetos com atributos do tipo classe devem destruir os atributos do tipo classe antes de se destruírem.



Agregação / Composição

- ▶ É o princípio que permite ao desenvolvedor considerar algo muito grande através do enfoque **Todo-Parte**.
- ▶ Todo-Parte também é conhecido como **Agregação**. Esse mecanismo permite a construção de uma classe agregada a partir de outras classes componentes. Usa-se dizer que um objeto da classe agregada (**Todo**) é **composta de** objetos das classes componentes (**Parte**) ou, objetos da classes componentes (**Partes**) **compõem** um objeto (**Todo**).



Classes Abstratas / Concretas

- ▶ Classes Abstratas
 - ▶ São classes para as quais não se pretende instanciar nenhum objeto;
 - ▶ Fornece uma superclasse apropriada da qual as outras classes (subclasses) podem herdar interface e/ou implementação.
- ▶ Classes Concretas
 - ▶ As classes da qual os objetos podem ser instanciados são chamadas classes concretas.



Polimorfismo

- ▶ Polimorfismo, significa muitas formas. Isto é, quando uma mesma operação pode assumir vários comportamentos.
- ▶ Um nome pode denotar objetos de muitas classes diferentes que estão relacionadas por alguma superclasse comum.
- ▶ Então, objetos denotados pelo mesmo nome tem a capacidade de responder a algum conjunto comum de operações de modos diferentes.



Sobreposição de Métodos

- ▶ É possível, ao definirmos uma classe como subclasse de outra, alguns métodos herdados não sirvam e necessitem de nova implementação.
- ▶ A sobreposição de métodos está diretamente associado ao conceito de Polimorfismo.



Métodos Abstratos

- ▶ Permite determinar os métodos que poderão ser implementados em classes descendentes, porém sem implementá-los na superclasse.
- ▶ Padroniza a interface das classes determinando os serviços que poderão ser implementados pelas subclasses.
- ▶ Um método abstrato é um método virtual que não é implementado na classe onde ele é declarado. Sua implementação é realizada em classes descendentes (subclasse).