

Conceitos Básicos de Orientação a Objetos

prof. Angelo C. Loula

1

Agenda

- Conceitos Básicos:
 - Objetos
 - Classes
 - Atributos e Estado
 - Métodos, Mensagens e Comportamentos
 - Encapsulamento
 - Herança
 - Polimorfismo e Ligação Dinâmica
 - Relacionamentos e Composição
 - Acoplamento e Coesão

2

Revisão

- O que são Objetos?
- O que são Classes?
- Qual a relação entre elas?
- Atributos? Métodos?
- Encapsulamento?
- Estado?
- Mensagens? Comportamento?

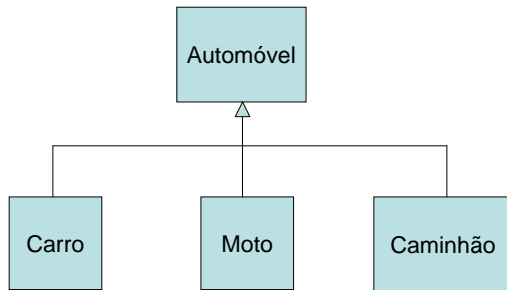
3

Herança

- Classes são uma abstração de um conjunto de objetos com métodos e atributos comuns
- Mas e se objetos tiverem somente alguns métodos e atributos comuns?
- Exemplo:
 - Classes Carro, Moto, Caminhão
 - Exemplos de objetos dessas classes?
 - Exemplos de atributos e métodos delas?

4

Herança



- Herança
 - Superclasses e subclasses (classe base e classe derivada)
 - Subclasse herda atributos e métodos da superclasse e estende com seus próprios atributos e métodos
- Pode haver superclasse de superclasse ou subclasse de subclasse, sem limite de herança

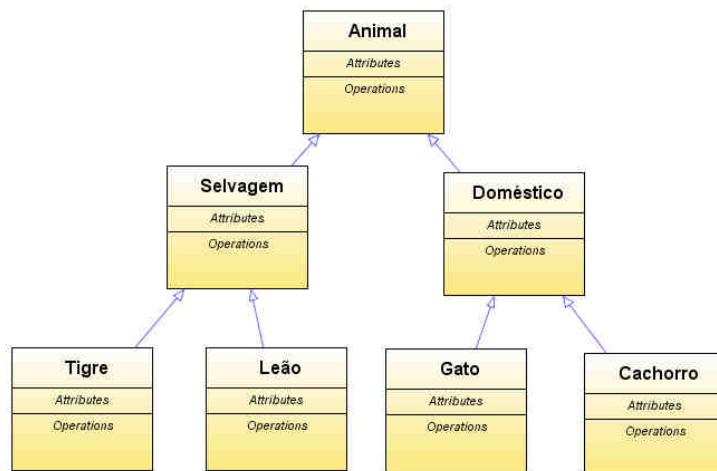
5

Herança

- Herança
 - Superclasse é uma generalização das subclasses
 - Subclasse é uma especialização da superclasse
- Relacionamento “é um”
- 2 Tipos:
 - Simples: Herança de apenas 1 superclasse
 - Múltipla: Herança de mais de 1 superclasse
- C++ suporta herança múltipla, enquanto java suporta apenas herança simples
 - Evita problemas de conflitos
 - Herança múltipla simulada com interfaces (estudada futuramente)
- Outros exemplos de herança?

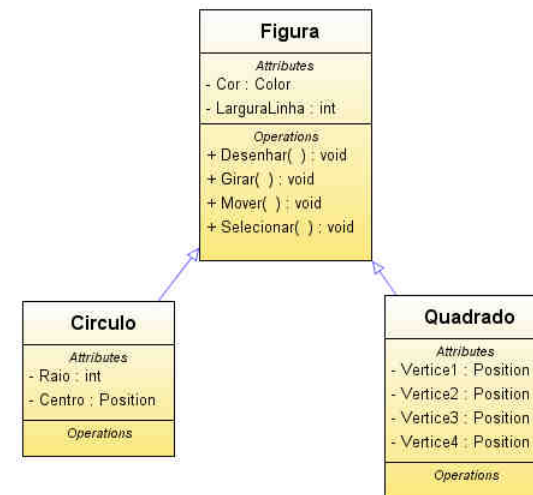
slide baseado em material do prof. Kellyton Brito 6

Herança: Exemplos



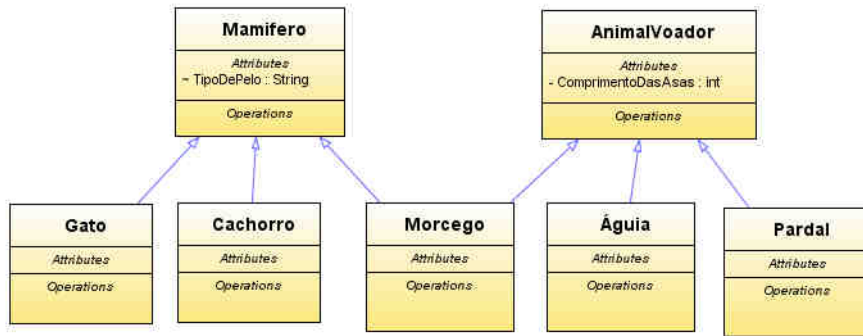
slide baseado em material do prof. Kellyton Brito 7

Herança Simples: Exemplo



slide baseado em material do prof. Kellyton Brito 8

Herança Múltipla: Exemplo



slide baseado em material do prof. Kellyton Brito 9

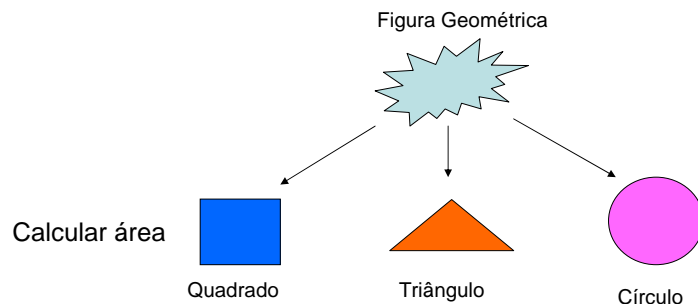
Polimorfismo

- do Grego: poli = muitas, morphos = formas
- Um método em uma classe pai pode ter várias implementações nas classes filhas
- Um mesmo método com uma mesma assinatura (protótipo) é definido para duas ou mais classes dentro da hierarquia de herança
- Mas os métodos polimórficos têm implementações diferentes em cada classe

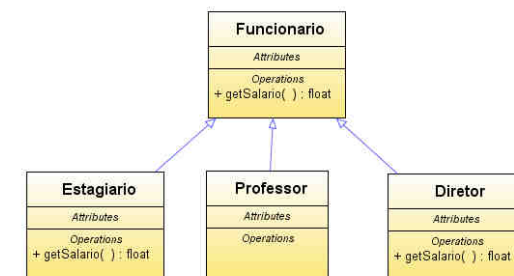
slide baseado em material do prof. Kellyton Brito 10

Polimorfismo

- Uma mesma operação pode atuar de diferentes modos em diferentes classes, definindo métodos específicos.



slide baseado em material do prof. Kellyton Brito 11



- Na implementação:
 - O do estagiário retorna meio salário
 - O do Diretor faz outro cálculo, de acordo com o desempenho da empresa
 - getSalario do funcionário (padrão) retorna um salário mínimo
 - O do professor herda do funcionário: salário mínimo

slide baseado em material do prof. Kellyton Brito 12

Polimorfismo

- Ligação Dinâmica vs Estática
 - Ligação dinâmica, adiada (dynamic, late binding)
 - Durante a execução, é determinado o método a ser chamado baseado no tipo real do objeto envolvido no momento, mesmo que seja referenciado pela superclasse.
 - Permite que um objeto referenciado pela sua superclasse, tenha seu método de subclasse invocado na execução.
 - Essencial para polimorfismo
 - Método é definido pelo **conteúdo**.
 - Ligação estática
 - Durante a compilação, é determinado o método a ser chamado pela referência (estática) do objeto e não pelo seu tipo real.
 - Somente é invocado o método da classe pela qual o objeto está sendo referenciado.
 - Sem polimorfismo!
 - Método é definido pela **referência, declaração**.

13

Sobrescrita de Método

- Conceito intimamente ligado ao polimorfismo
- Quando um método de uma classe pai é reescrito na classe filha
 - O método da classe pai é **sobrescrito** e “esquecido” na classe filha
- Métodos devem ser **idênticos**: Nome, Atributo e Retorno iguais

slide baseado em material do prof. Kellyton Brito 14

Sobrecarga de método

- Utilização do mesmo método mas com assinaturas diferentes
 - Diferente de sobrescrita (mesma assinatura)
- Torna o código mais legível
- Torna o desenvolvimento mais natural
- Permite abstrações de tipos
 - Desenvolvedor não precisa saber os tipos exatos
- Exemplo:
 - somar inteiros ou somar reais
 - consertar carro e consertar moto

slide baseado em material do prof. Kellyton Brito 15

Relacionamento / Associação

- Os objetos do mundo real se relacionam entre si
- Igualmente, os objetos do sistema se relacionam entre si
 - Aluno* senta na *Cadeira*
 - Professor* tem um *Laptop*
- Como modelar esse relacionamento?

slide baseado em material do prof. Kellyton Brito 16

Relacionamento / Associação

- Relacionamento próximo a posse
 - Caracterizado pela expressão “tem um” ou “tem alguns”

Aluno tem um Pai

Aluno tem alguns Cadernos

Pai tem um Filho

SalaDeAula tem algumas Cadeiras

Corpo tem algumas Células

País tem um ChefeDeEstado

slide baseado em material do prof. Kellyton Brito 17

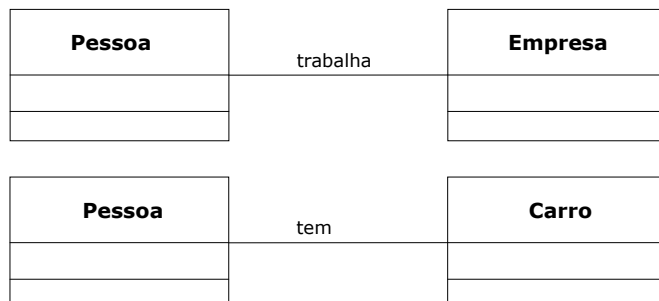
Relacionamento / Associação

- Tipos de Relacionamento
 - Associação Simples
 - Agregação
 - Composição

slide baseado em material do prof. Kellyton Brito 18

Associação Simples

- Relação estrutural entre Classes
- Representada por uma linha simples
 - Ex.: Uma Pessoa trabalha em uma Empresa
 - Uma Pessoa tem um Carro



slide baseado em material do prof. Kellyton Brito 19

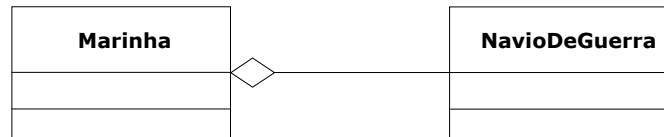
Associação Simples

- Os objetos são independentes
- Um objeto pode existir sem a necessidade do outro
- Os objetos podem ser utilizados de forma independente em outras partes da aplicação

slide baseado em material do prof. Kellyton Brito 20

Agregação

- Forma especializada de Associação
- Relacionamento Todo – Parte
- Conhecida como relação de Conteúdo
 - Uma das classes é uma parte ou está contida em outra
- Representado por um losango vazio



slide baseado em material do prof. Kellyton Brito 21

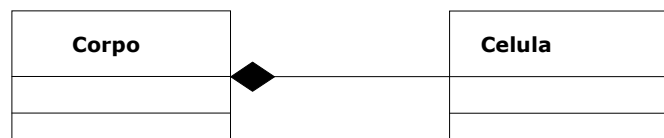
Agregação

- Objetos ainda existem sem a necessidade do outro
- Mas não fazem muito sentido sozinhos
- Exemplo:
 - Uma Reunião é o agregado de uma Pauta, uma Sala e Participantes, que existirão independente de existir a reunião. Porém uma pauta não faz sentido se não for para uma reunião

slide baseado em material do prof. Kellyton Brito 22

Composição

- Relação de posse mais forte
- Agregação em que uma parte constitui a outra
- A parte não vive sem o todo
- Representado por um losango preenchido



slide baseado em material do prof. Kellyton Brito 23

Associação, Agregação, Composição

- Dicas:
 1. Se eu deletar A, terei que deletar B?
 - Sim: Composição
 - Ex.: Pedido e compras. Uma compra sempre faz parte de um pedido.
 - Não: Vá para pergunta 2
 2. O Objeto B tem alguma utilidade sozinho?
 - Sim: Associação
 - Não: Agregação
 - Ex.: Carro e rodas. Uma roda existe sem um carro, mas ela sempre serve para ser colocada em alguma coisa (carro, moto, árvore-balanço)

slide baseado em material do prof. Kellyton Brito 24

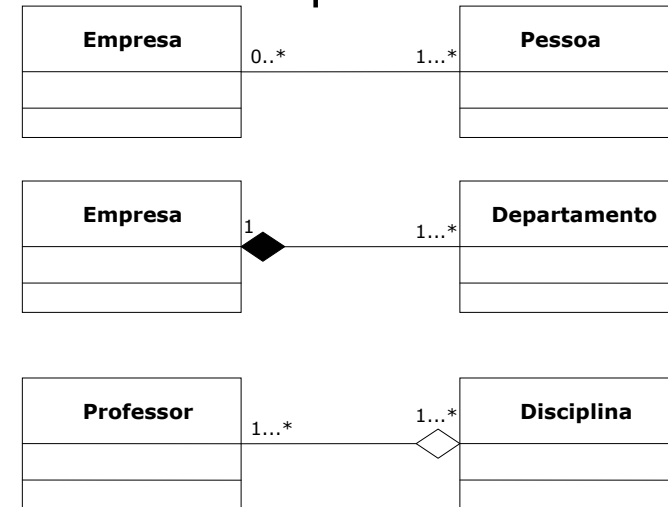
Multiplicidade

- Multiplicidade define quantos objetos participam do relacionamento
 - Depende do sistema a ser modelado
- É especificado em cada uma das pontas da associação
- Usa-se 0, 1 e * (ou n)



slide baseado em material do prof. Kellyton Brito 25

Multiplicidade



slide baseado em material do prof. Kellyton Brito 26

Modelando um Sistema

- Como modelar um sistema, efetivamente?
 - Normalmente Substantivos viram Classes
 - Estados e Adjetivos (propriedades) viram Atributos
 - Verbos (ações) viram Métodos
- Exemplo: Modelando uma biblioteca

slide baseado em material do prof. Kellyton Brito 27

Modelando uma biblioteca

- Substantivos – Classes
 - Biblioteca
 - Livros
 - Revistas
 - Cliente
 - Reserva
 - Empréstimo
 - Funcionário

slide baseado em material do prof. Kellyton Brito 28

Modelando uma biblioteca

- Livro ou Revista
 - Atributos:
 - Nome, Editora, Edição, Ano, outros
 - Métodos:
 - (manipulação básica dos atributos)
- Funcionário
 - Atributos:
 - Nome, CPF, Matrícula, Cargo.
 - Métodos:
 - CadastrarLivro, ConsultarLivro, RegistrarEmprestimo,, CadastrarCliente, ConsultarCliente, RemoverCliente

slide baseado em material do prof. Kellyton Brito 29

Modelando uma Biblioteca

- Cliente
 - Atributos:
 - Nome, Matrícula, CPF, Endereco
- Reserva
 - Atributos:
- Empréstimo
 - Atributos:

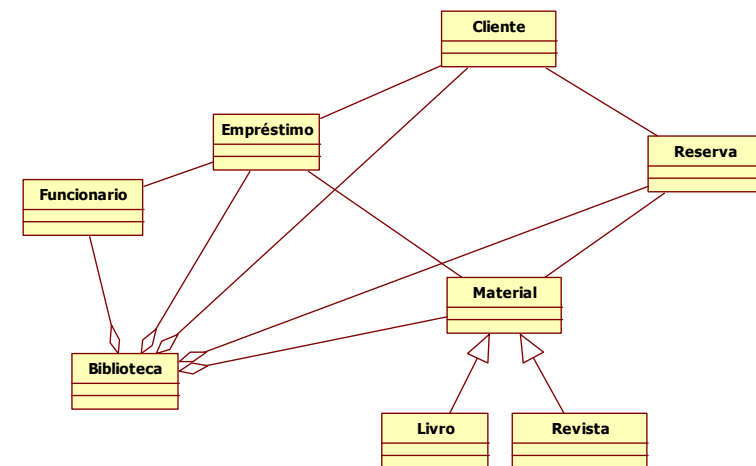
slide baseado em material do prof. Kellyton Brito 30

Modelando uma biblioteca

- Por tratar-se de uma biblioteca, a classe biblioteca será a “Gerente” da aplicação, contendo a chamada principal, a interface gráfica e os outros objetos.
- Biblioteca
 - Atributos: ListaDeLivros, Lista de Revistas, ListaDeFuncionarios, ListaDeClientes, ListaDeReservas, ListaDeEmpréstimos
 - Métodos: Inicializar, LogarFuncionario, MostrarMenu, (manipulação das listas)

slide baseado em material do prof. Kellyton Brito 31

Modelando uma Biblioteca



32

Acoplamento e Coesão

- Ao construirmos um sistema, devemos ter sempre como meta diminuir o acoplamento e aumentar a coesão
- Coesão: medida de quão focada é uma classe, relativa a diversidade de coisas que ela faz
 - Exemplo: classe de conta com informações sobre o correntista: pouco coesa
 - conta conjunta? várias de contas de um correntista? especialização de conta (pf/pj ou cc/poupança)?

33

Acoplamento e Coesão

- Acoplamento: grau de dependência entre as partes do sistema, do funcionamento interno de outras partes, quanto uma classe conhece das demais
 - Tipos
 - Identidade: classe como atributo
 - Representação: referência a classe e seu funcionamento
 - Subclasses: relação pelas subclasses
 - Herança: subclasse herda da superclasse

34

Introdução

- Erros do programador procedural em orientação a objetos
 - Pouca (capacidade de) reutilização
 - Controle de execução do método pelos parâmetros (ao invés de sobrecarga)
 - Método igual em subclasse mas sem polimorfismo