



Universidade Estadual de Maringá
Departamento de Informática



Programação Orientada a Objetos

Conteúdo baseado nos materiais dos Professores:
Marcos Aurélio Domingues (DIN/UEM)
Edson Oliveira (DIN/UEM)
Bruno Boniati (UFMS)

Prof.^a Juliana Keiko Yamaguchi
janeiro de 2019

Objetivos

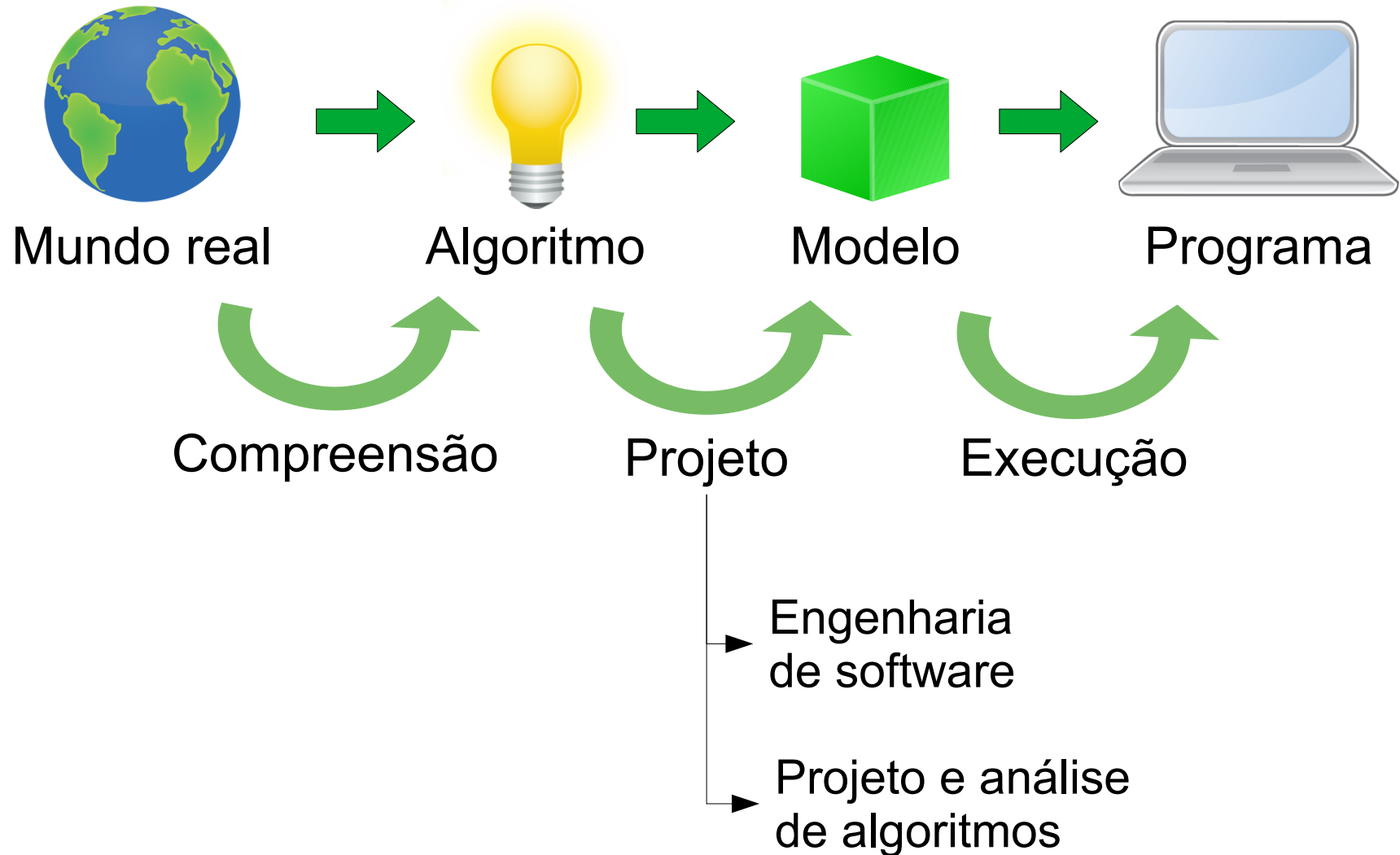
- Apresentar uma introdução ao paradigma de programação orientada a objetos.
- Discutir o conceito de abstração.
- Estudar o conceito de classes e objetos.

Contextualização

- Como resolver um problema por meio de um computador?
- Quais os paradigmas de desenvolvimento existentes?

Contextualização

Como construir software?



Contextualização

Paradigmas de programação

- Paradigma procedural ou estruturado
 - Sequência de instruções executadas do início ao fim, eventualmente organizado em sub rotinas (subprogramas).
- Paradigma orientado a objetos
 - Representação de objetos que trocam mensagens entre si, com o objetivo de realizar uma tarefa.
- Paradigma funcional e lógico
 - Declaração de regras e uso de funções para encontrar um conjunto solução para o problema.

Paradigma orientado a objetos

Conceitos básicos

- O paradigma orientado a objetos é fundamentado no conceito de **classes** e **objetos** que enviam **mensagens** entre si para realizar uma tarefa ou resolver um problema.
- Todo o problema tem necessidade de ser resolvido por meio desse paradigma?

Paradigma orientado a objetos

É sempre necessário?

- Nem sempre...
- Há situações em que a tarefa a ser executada é tão simples que pode ser resolvido pelo paradigma estruturado.
- Exemplo:
 - Fazer um programa que dada a entrada, forneça a resposta de acordo com a equação:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Paradigma orientado a objetos

É sempre necessário?

- No entanto, para aplicações mais complexas, o paradigma orientado a objetos pode trazer benefícios.
- Exemplo: Construção de interface gráfica de usuário.
 - Como desenvolver uma tela de sistema de software considerando:
 - Paradigma estruturado;
 - Paradigma orientado a objetos.



Paradigma orientado a objetos

Conceitos básicos

- O que são objetos?
- O que são classes?
- Como definir classes e objetos?
- O princípio básico da orientação a objetos é a abstração.
 - O que é abstração?

Abstração

- Abstração:
 - Ato ou efeito de abstrair.
- Abstrair:
 - Separar, considerar isoladamente coisas que se encontram unidas;
 - Simplificar um conceito ou entidade para facilitar seu entendimento.

Abstração

- No contexto da computação, abstração é a capacidade de compreender um problema do mundo real, que envolve entidades reais, propondo uma representação simplificada para solucionar um problema específico.
- Exemplo: Cadastro de usuários.
 - Qual a finalidade?
 - Quais dados são importantes?

Abstração

- Portanto, abstração é a habilidade de se concentrar nos aspectos essenciais de um contexto qualquer, ignorando características menos importantes ou detalhes operacionais.
- No paradigma orientado a objetos, a abstração é dada por meio de **classes** e **objetos**.

Classe

- A classe é a unidade básica de trabalho em um programa orientado a objetos.
- Representa um modelo abstrato a partir do qual são criadas instâncias (objetos).
- Uma classe representa uma coleção de objetos que possuem características em comum.

Classe

Exemplo

- Classe de trabalhadores
 - Quais as características comum dos trabalhadores?
 - Quais as funções que todo trabalhador deve realizar?
- Classe de estudantes
 - Quais as características comum dos estudantes?
 - Quais as funções que todo estudante deve realizar?

Objeto

- Objeto é uma instância de uma classe.
 - Possui características com valores reais ou factuais.
- O objeto realiza as ações definidas pela sua classe.
- Em outras palavras:
 - O comportamento do objeto é definido na classe.
 - O objeto realiza o comportamento.

Classe vs. Objeto

- O objeto está para classe da mesma forma que a variável está para o tipo de dado.
- Exemplo:
 - Suponha que o programa tem que manipular as informações de um carro.
 - Como você representaria a entidade carro:
 - No paradigma estruturado;
 - No paradigma orientado a objetos.

Classe

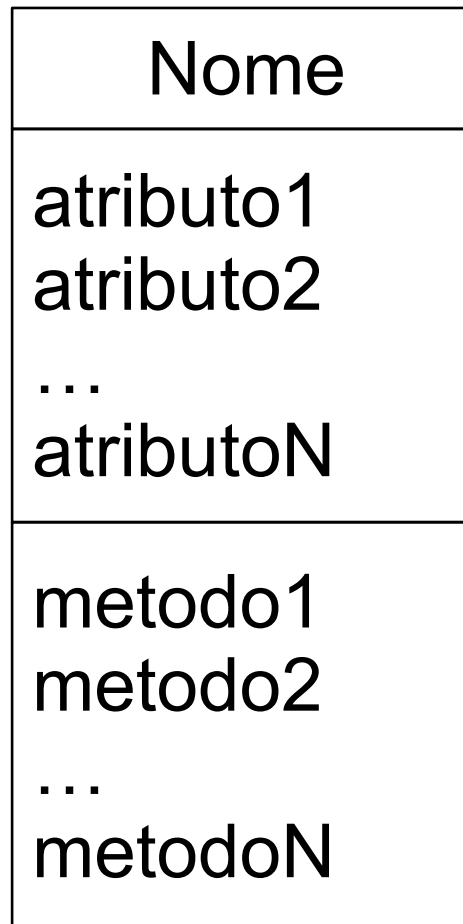
Características principais

- Atributos
 - São os dados (simples ou compostos) que caracterizam objetos daquela classe;
 - São armazenados em variáveis;
 - Constituem o estado do objeto.
- Métodos
 - São as operações que manipulam os dados, além de desempenhar outras ações do objeto.
 - Determinam o comportamento dos objetos (o que eles podem fazer) no sistema.

Classe

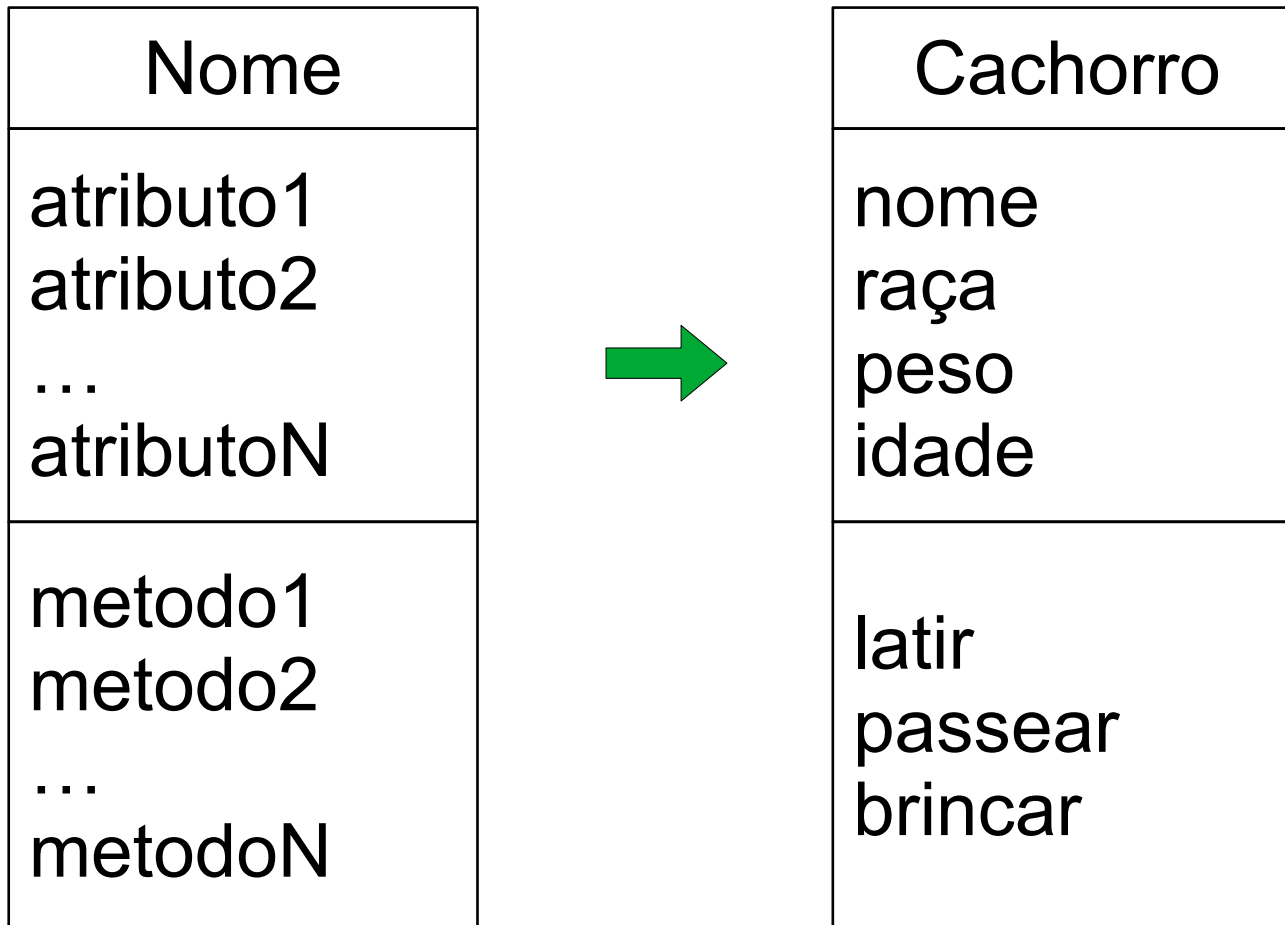
Representação gráfica

- Modelo de classe definido pela UML (*Unified Modeling Language*)



Classe

Representação gráfica – Exemplo



Classe vs. Objeto

Representação gráfica – Exemplo

Cachorro
nome raça peso idade
latir lassear brincar



Classe vs. Objeto

Exercícios

- Determine as classes principais para os tipos de sistema a seguir:
 - um sistema acadêmico;
 - um sistema para biblioteca.
- Determine os principais atributos e possíveis métodos de cada classe.

Atividade

- Leia o artigo “Os 4 pilares da programação orientada a objetos”.
 - <https://www.devmedia.com.br/os-4-pilares-da-programacao-orientada-a-objetos/9264>
 - Acessado em março de 2019.

Atividade

- Responda às seguintes perguntas:
 - No contexto da programação orientada a objetos, o que significa cada classe ter seus próprios dados e métodos, ao contrário da programação estruturada, na qual os procedimentos são globais?
 - Explique com suas próprias palavras, as motivações para usar o paradigma orientado a objetos em vez do paradigma estruturado (ou procedural)