

# Introdução

Prof. Hugo de Paula



PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS  
Departamento de Ciência da Computação

# Sumário

- 1 Programação modular
- 2 Modelo orientado para objetos
- 3 Histórico



# Programação modular

- **Programação modular**

- Técnica de projeto (*design*) de software.
- Cada módulo é uma unidade independente que provê uma funcionalidade específica de um programa.
- Módulos se comunicam.
- Programação em larga escala: modelo estruturado ou orientado para objetos.
- Decomposibilidade de software.



# Modelo orientado para objetos

- Objeto possui (*Cay S. Horstmann, Object-Oriented Design & Patterns, 2006.*):
  - estado;
  - comportamento;
  - identidade.
- A Orientação para objetos abstrai o mundo real utilizando objetos que interagem entre si.
- Utiliza o princípio da decomposibilidade para desenvolver sistemas modulares.
- Melhora a reusabilidade e extensibilidade dos softwares.



# Modelo orientado para objetos

- **Programação Orientada para Objetos (OOP):** Método de implementação onde os programas são organizados como coleções de objetos cooperativos, cada objetos representando uma instância de uma classe.
- **Projeto Orientada para Objetos (OOD):** Método de projeto onde o processo da decomposibilidade orientada para objetos é utilizada e modelos físicos e lógicos de objetos são descritos.
- **Análise Orientada para Objetos (OOA):** Método de análise que examina os requisitos de um sistema de uma perspectiva de classes e objetos, encontrados no vocabulário o domínio do problema.



# Alan Kay, pai da programação Orientada para Objetos

*I'm sorry that I long ago coined the term “objects” for this topic because it gets many people to focus on the lesser idea.*

Para *Alan Kay*, POO significa apenas:

- troca de mensagens,
- proteção e retenção local e ocultamento do estado ou processo, e
- associação tardia e dinâmica de tudo o que for possível.

A grande ideia é “troca de mensagens”

A chave para desenvolver sistemas extensíveis de qualidade é muito mais projetar como os módulos se comunicam do que como suas propriedades internas e seus comportamentos deveriam ser.



# História da programação orientada para objetos

- Conceito de classe, objeto e herança (década de 60):  
linguagem **Simula 67**.

## Alo Mundo em Simula 67

```
begin
  OutText("Hello , world!");
  OutImage
end
```



## Exemplo de programa em Simula 67

```
begin
  class Vehicle;
    virtual: procedure sound is procedure sound;;
  begin
    end;
  Vehicle class Car;
  begin
    procedure sound;
    begin
      OutText("Beep beep!");
      OutImage;
    end;
  end;

  ref (Vehicle) array vehicles (1 : 2);
  Integer i;
  vehicles (1) := new Car;   vehicles (2) := new Car;
  for i := 1 step 1 until 2 do vehicles(i).sound
  end;
```





# Smalltalk

- **Smalltalk**: primeira linguagem orientada para objetos completa (década de 70).
  - tipos abstratos de dados (classes e objetos)
  - herança
  - amarração dinâmica (polimorfismo)
  - design de interface gráfica
- Desenvolvida por Alan Kay, Dan Ingalls e Adele Goldberg.
- Em Smalltalk tudo é objeto: números, classes, métodos, blocos de código, etc.

## Alo Mundo em Smalltalk

```
'Hello World!' printNL
```



# História da programação orientada para objetos

- outros exemplos: Eiffel, Java, Object Pascal, C#, Python.
- linguagens como C++, Ada e Perl suportam o conceito de classe mas não são consideradas LPs OO em um sentido mais rigoroso.