

Programação Orientada à Objetos

Fundamentos

Prof. Edson Alves

Faculdade UnB Gama

2020

Sumário

1. **Programação Orientada a Objetos**
2. *Small Talk*

Introdução

- ▶ A **programação orientada a objetos** é um paradigma de programação baseada em objetos
- ▶ **Objetos** são estruturas de dados que armazenam informações (**atributos**) e procedimentos (**métodos**)
- ▶ Os programas consistem em interações entre os diferentes objetos
- ▶ Estas interações são conduzidas por meio de trocas de **mensagens**
- ▶ Os objetos são instâncias de **classes**, as quais determinam os tipos, atributos e métodos dos objetos
- ▶ Muitas linguagens multiparadigmas suportam a orientação a objetos
- ▶ Algumas outras, como SmallTalk, suporta apenas o paradigma da programação orientada a objetos

Histórico

- ▶ O termo objeto, no contexto de programação orientada a objetos, surgiu no MIT no final dos anos 1950 e no início dos anos 1960
- ▶ Ele se referia a um item identificado com atributos
- ▶ Também no MIT, Iva Sutherland desenvolveu um software denominado Sketchpad, ancestral dos programas CAD, e definiu os termos objeto e instância
- ▶ A linguagem Simula67 introduziu o conceito formal de objetos em programação, e também a noção de classes
- ▶ A expressão “programação orientada a objetos” foi introduzida pela linguagem SmallTalk, nos anos 1970
- ▶ Esta linguagem faz uso de objetos e mensagens como base de suas computações, e as classes podiam ser modificadas dinamicamente
- ▶ Nos anos 1990 este paradigma se tornou a metodologia dominante de desenvolvimento de software, obtendo suporte na maior parte das linguagens então existentes

Visão geral

- ▶ Além de unir dados e funções, os objetos também são **tipos de dados abstratos**, isto é, são definidos por suas interfaces, e não por suas implementações
- ▶ Os objetos também oferecem suporte para **polimorfismo** e para **herança**
- ▶ Os objetos são organizados em hierarquias de **classes**
- ▶ As classes e as **subclasses** são análogos aos conjuntos e subconjuntos da matemática
- ▶ Os objetos são **modulares**, no sentido que são responsáveis por seus dados e pelo próprios comportamentos
- ▶ Esta característica é denominada **encapsulamento**

Visão geral

- ▶ Na programação orientada a objetos os dados devem, sempre que possível, ser protegidos de acessos diretos
- ▶ O acesso aos dados do objetos deve ser feito por meio dos métodos de sua **interface**
- ▶ Outro aspecto importante deste paradigma é o **reuso** de código
- ▶ São duas as principais formas de reuso: construir novos objetos através da **composição** de objetos já existentes ou estender um objeto por meio de herança
- ▶ Um objeto (ou módulo) está **aberto** se ele permite sua extensão por herança
- ▶ Ele está **fechado** se sua interface é estável e bem definida, de modo que não pode (ou deve) ser estendida

Referências

1. **SHALOM**, Elad. *A Review of Programming Paradigms Throughout the History – With a Suggestion Toward a Future Approach*, Amazon, 2019.