



**Attribution-NonCommercial-
NoDerivatives 4.0 International
(CC BY-NC-ND 4.0)**



Este trabalho está licenciado com uma Licença [Creative Commons -
Atribuição-NãoComercial-SemDerivações 4.0 Internacional](https://creativecommons.org/licenses/by-nc-nd/4.0/).

Programação Orientada a Objetos

2

BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO

PROF. LUCIO AGOSTINHO ROCHA

AULA 1: INTRODUÇÃO

1º.SEMESTRE 2023

Sejam Bem-Vindos!

Introdução

4

- Apresentação do professor
- Disciplina: Programação Orientada a Objetos – POOCO4A
- Turmas: COM4A - 2023/1

Introdução

5

- **Objetivos:**
 - O aluno será capaz de desenvolver sistemas baseados nos conceitos formais da Orientação a Objetos, compreender os paradigmas da programação orientada a objetos e implementar estudos de caso em Linguagem de Programação Orientada a Objetos.

Introdução

6

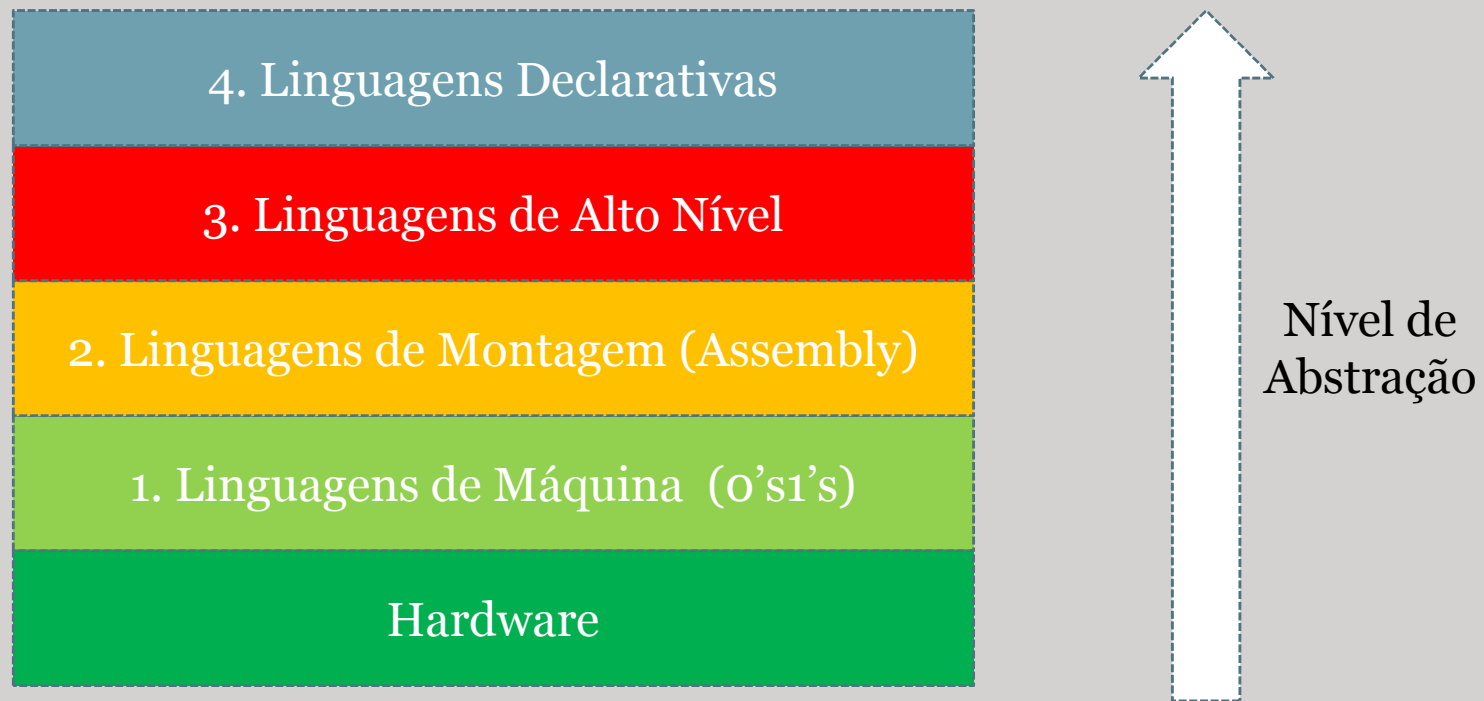
- **Planejamento**
 - Plano de Ensino e Ementa
 - Sistemas de avaliação
 - Previsão de datas
- **Plataforma Classroom**
 - ✦ Material e trabalhos da disciplina
 - ✦ Faça a sua inscrição na disciplina
 - ✦ Grupo de mensagens

Breve Histórico: Programação Orientada a Objetos

Breve Histórico: Programação Orientada a Objetos

8

- Ao longo dos anos, o ciclo de desenvolvimento de software passou por muitas mudanças.



Breve Histórico: Programação Orientada a Objetos

9

- A Programação Orientada a Objetos (POO) é uma metodologia de desenvolvimento de software que busca:
 - Reduzir a Complexidade do software: formar abstrações da aplicação em termos de “objetos” e “classes”;
 - Reusabilidade:
 - ✦ Reutilizar componentes de software facilita o rápido desenvolvimento.
 - ✦ Utilizar outros componentes ao invés de criar novos.
 - ✦ Herdar bons componentes ao invés de “reinventar a roda”
 - Simplificar a criação de interfaces amigáveis “WYSIWYG” (What You See Is What You Get): interfaces gráficas, janelas, menus e ícones são vistos como “objetos”;

Breve Histórico: Programação Orientada a Objetos

10

- O termo “orientado a objetos” surgiu da linguagem Smalltalk (Xerox Palo Alto, década de 70). Smalltalk teve forte influência das linguagens Simula e LISP.
- Diferentes definições de POO:
 - 1) Capretz[1]: POO é aquela que utiliza objetos. Um objeto é o nome de um tipo de dado abstrato instanciado a partir de uma classe. Um objeto possui variáveis privadas e procedimentos locais.
 - 2) Capretz[1]: POO é aquela que utiliza objetos de um determinado tipo. O objeto pode estar relacionado a outros objetos através de relações de subtipo e supertipo definidas por suas classes.

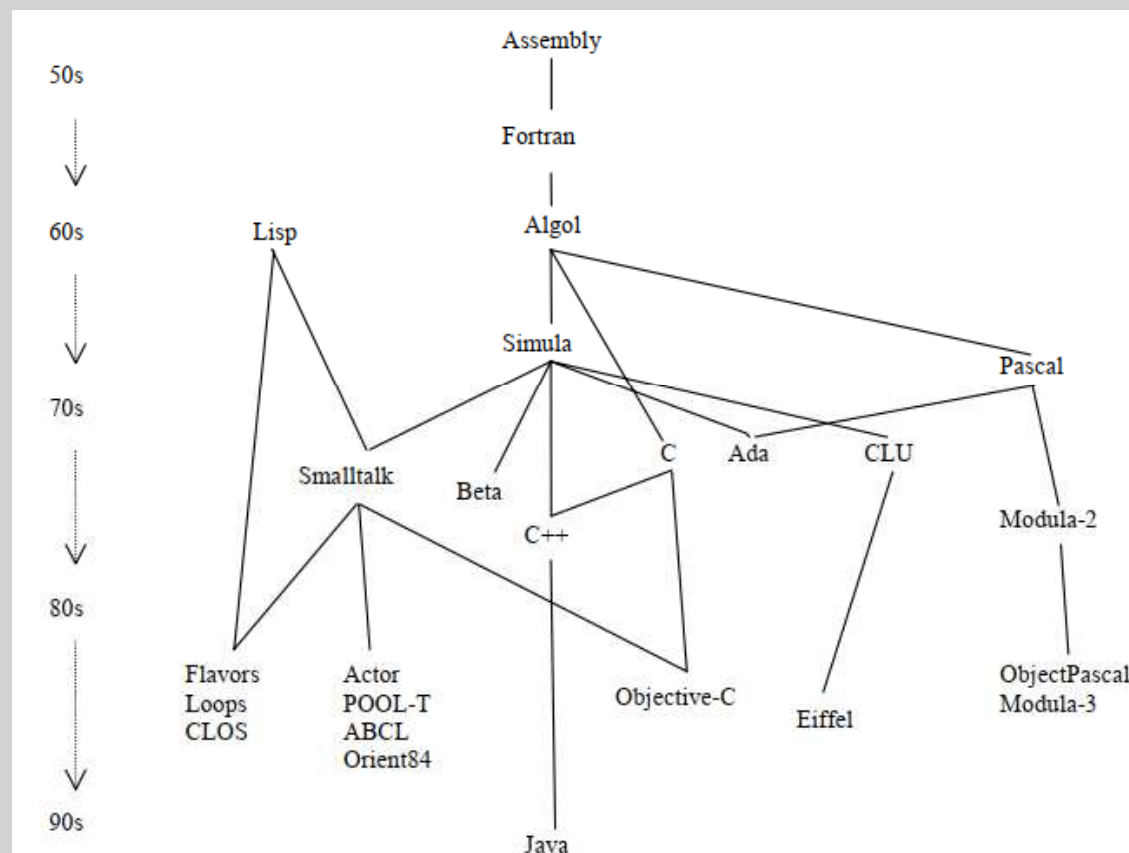
Breve Histórico: Programação Orientada a Objetos

11

- ...Diferentes definições de POO:
 - 3) Rentsch[6]: POO é aquela que utiliza objetos cujos atributos (propriedades) não são visíveis fora do objeto. Toda comunicação entre objetos é feita com passagem de mensagens. Todo processamento é realizado dentro dos objetos. O compartilhamento de propriedades entre os objetos é feito por herança.
 - 4) Wegner[8]: POO é aquela que utiliza objetos como entidades autônomas que possuem um estado e respondem a mensagens. Classes agrupam objetos que têm atributos e operações em comum. Herança é utilizada para organizar as classes de acordo com características comuns.

Breve Histórico: Programação Orientada a Objetos

12



- Figura - Evolução das Linguagens de Programação (Fonte: CAPRETZ, 2003)

Programação Orientada a Objetos

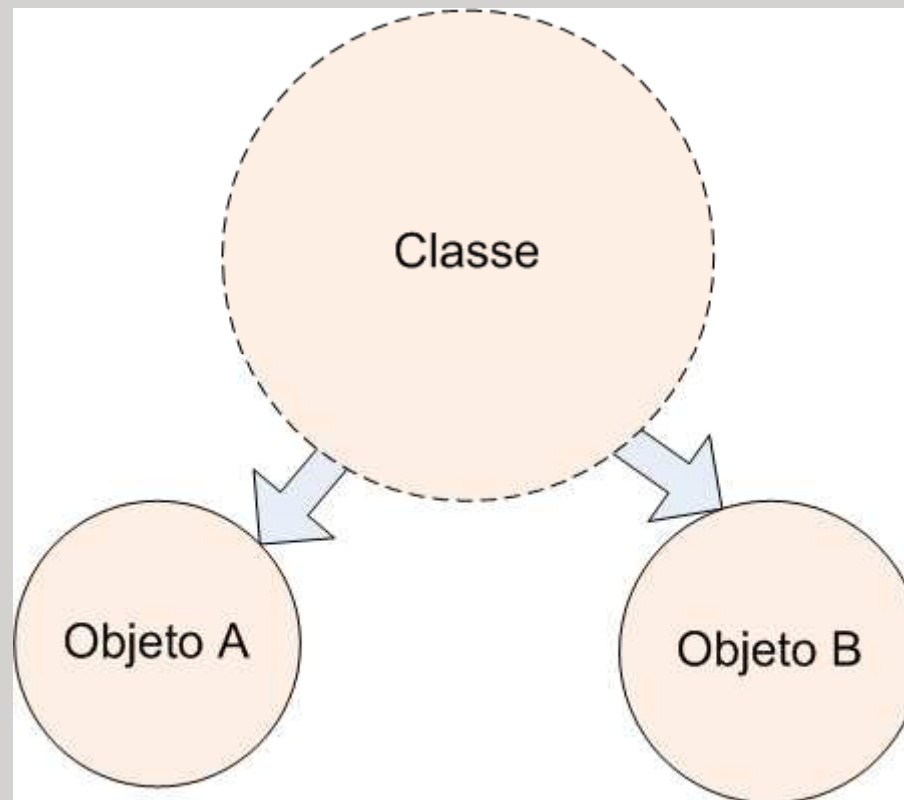
Programação Orientada a Objetos

14

- A Programação Orientada a Objetos (POO) é um tipo de programação em linguagem de alto nível.
- A unidade de construção de programas é a classe.
- A classe abstrai os aspectos mais relevantes do problema.
- A classe é similar a uma “planta de prédio”: a partir da mesma planta, podem ser criados prédios diferentes.

Programação Orientada a Objetos

15



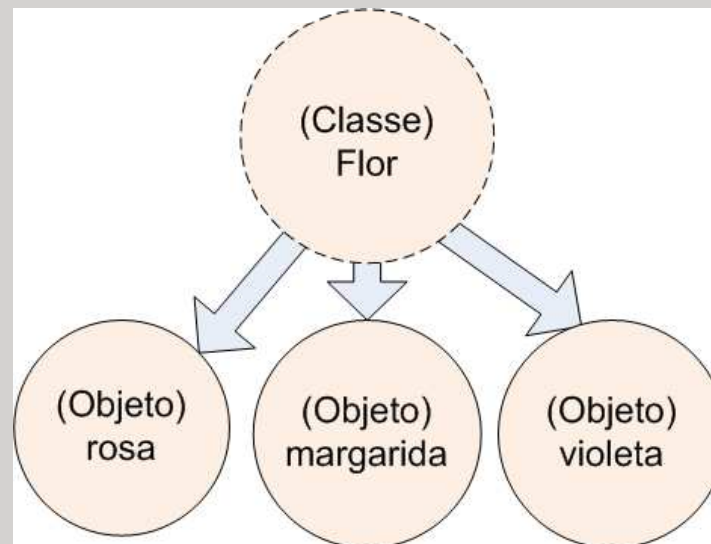
- Figura - Representação Simplificada de Classes e Objetos.

Programação Orientada a Objetos

16

- ✦ Exemplos:

- ✦ Flor: existem vários tipos de flores.
 - ✦ “rosa” (objeto) é uma instância de “Flor” (Classe)
 - ✦ “margarida” (objeto) é uma instância de “Flor” (Classe)
 - ✦ “violeta” (objeto) é uma instância de “Flor” (Classe)



- Figura - Representação Simplificada de Classes e Objetos.

Programação Orientada a Objetos

17

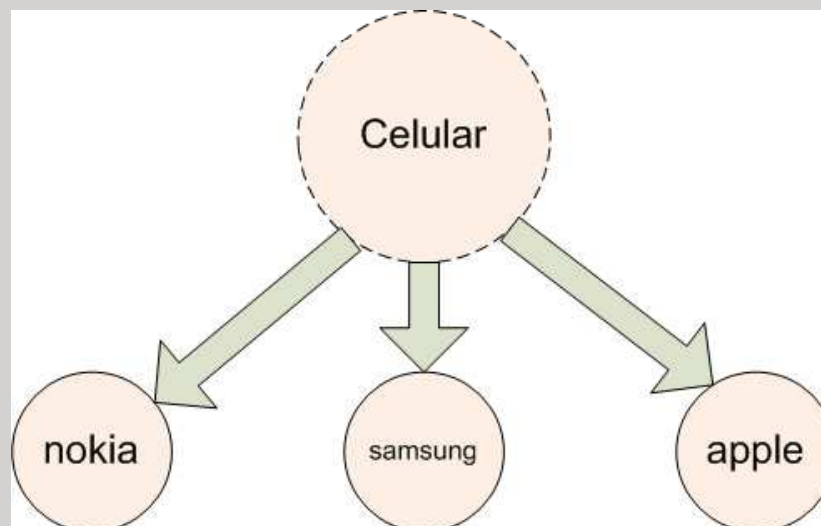
- ✦ Exemplos:

- ✦ Celular: existem vários tipos de celulares.

- ✦ “nokia” (objeto) é uma instância de “Celular” (Classe)

- ✦ “samsung” (objeto) é uma instância de “Celular” (Classe)

- ✦ “apple” (objeto) é uma instância de “Celular” (Classe)



- Figura - Representação Simplificada de Classes e Objetos.

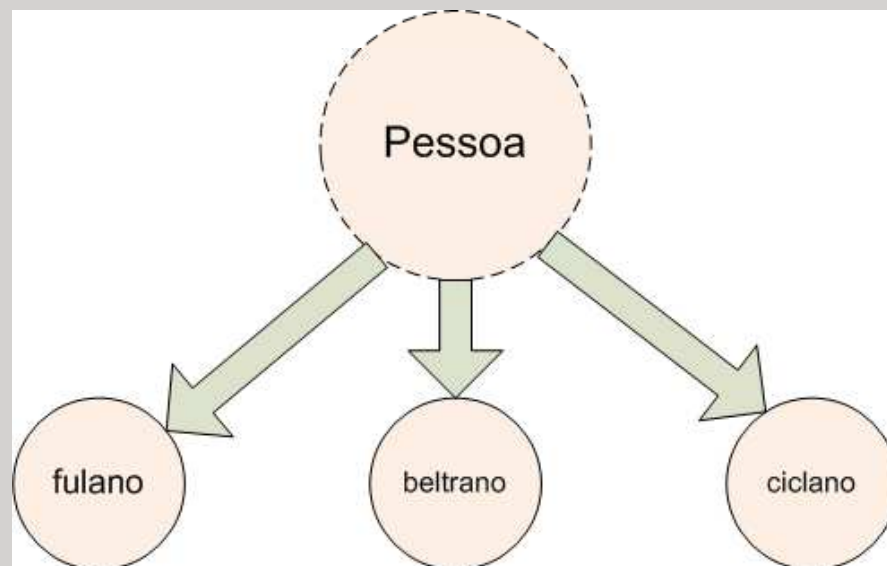
Programação Orientada a Objetos

18

- ✦ Exemplos:

- ✦ Pessoa: existem vários tipos de pessoas.

- ✦ “fulano” (objeto) é uma instância de “Pessoa” (Classe)
 - ✦ “beltrano” (objeto) é uma instância de “Pessoa” (Classe)
 - ✦ “ciclano” (objeto) é uma instância de “Pessoa” (Classe)



- Figura - Representação Simplificada de Classes e Objetos.

Programação Orientada a Objetos

19

- Definição do termo “classe”:
 - Classe: é um modelo que descreve objetos.
 - Classe é um modelo que define os atributos e métodos (comportamentos/ações) comuns para os seus objetos.
 - Classe: é um modelo abstrato que define as propriedades e comportamentos comuns aos objetos da classe.
 - Classe é uma unidade de programação estática.

Programação Orientada a Objetos

20

- Definição do termo “objeto” :
 - Objeto é uma instância de uma classe.
 - Objeto possui variáveis privadas (atributos)
 - Objeto possui procedimentos locais (métodos)
 - Propriedades e Comportamentos são armazenados em objetos:
 - ✦ Propriedades: são atributos do objeto.
 - ✦ Comportamentos: são as ações/operações que o objeto realiza.
 - Objetos: devem ser capazes de interagir através de seus métodos.
 - Objeto é uma unidade de programação dinâmica (possui ciclo de vida).

Motivação e Uso da Tecnologia

Motivação e Uso da Tecnologia

22

- Características da POO:
 - Encapsulamento: “objeto” protege os seus atributos (propriedades) e oculta a implementação de seus métodos. Os atributos e métodos são acessíveis através de:
 - ✦ Modificadores de acesso
 - ✦ Métodos acessores
 - ✦ Métodos mutadores.
 - Ocultamento: atributos e métodos são visíveis apenas dentro do contexto do objeto.
 - Fraco acoplamento: objeto possui apenas as informações específicas do seu contexto.
 - Modularidade: a aplicação é dividida em partes menores (“objetos”) que contêm informações específicas (estado) para determinado propósito.
 - Comunicação via Interfaces: objetos se comunicam por troca de mensagens através de suas interfaces.

Motivação e Uso da Tecnologia

23

- Programas Orientados a Objetos são mais fáceis de entender, corrigir e modificar.
- Objetos: componentes de software reutilizáveis.
- Objetos: possuem estado (propriedades: atributos) e comportamentos (ações: métodos)
- Classes: modelos genéricos que definem atributos e comportamentos comuns para instanciar objetos.
- Classes têm relação semântica:
 - Herança: permite o reaproveitamento de código.

Linguagem de Programação Java

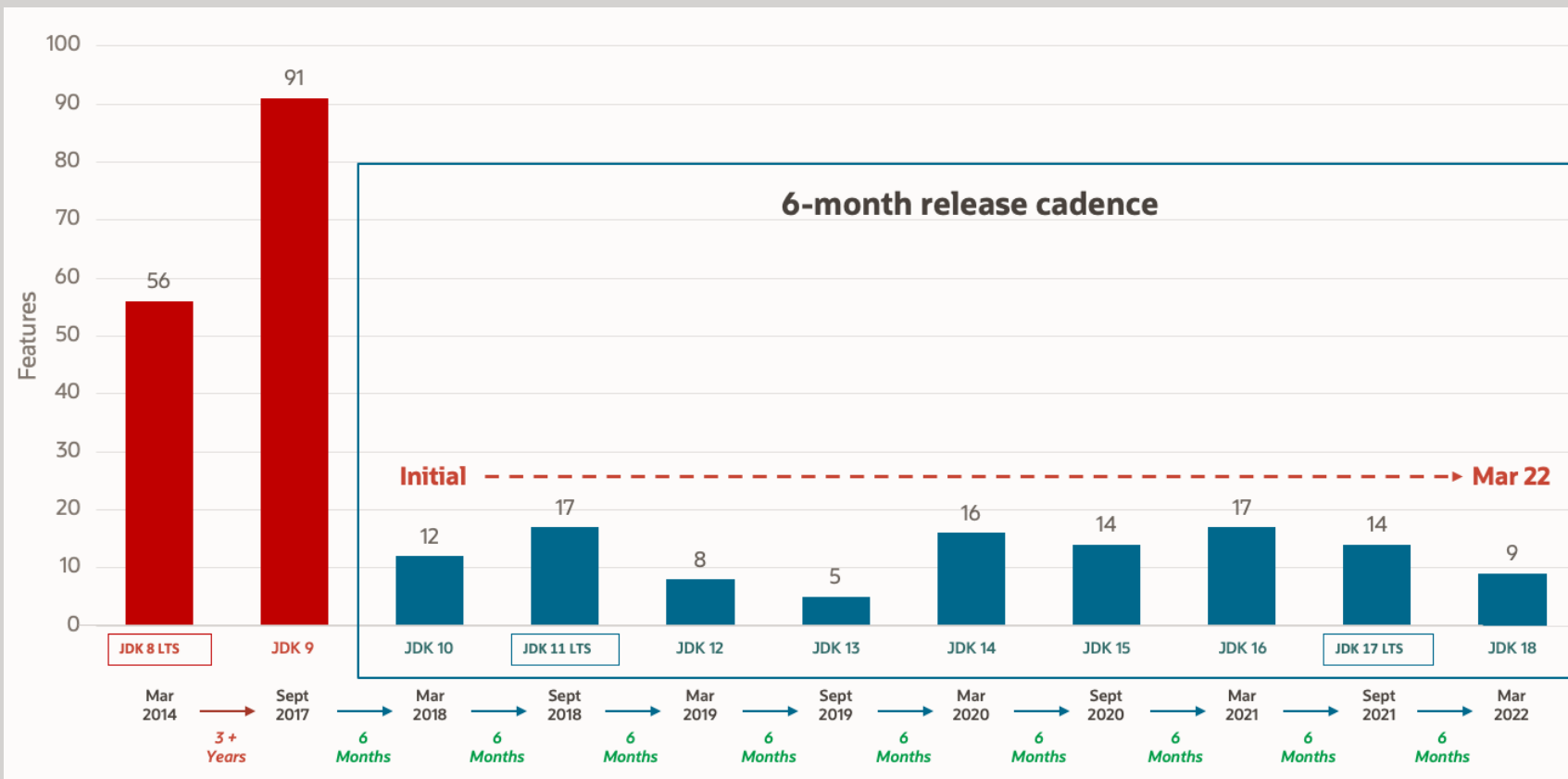
Linguagem de Programação Java

25

- Java é uma linguagem de programação orientada a objetos
- Java é baseada nas linguagens C e C++
- James Gosling (Sun Microsystems, 1991). A Sun Microsystems foi adquirida pela Oracle em 2010.
- Conteúdo dinâmico para a Web.
- Atualmente, mais de 8 bilhões de dispositivos (de software embarcado a sistemas de computação de alto desempenho)
- Ecossistema:
 - Java Runtime Edition (JRE): apenas para a execução de programa Java.
 - Java Development Kit (JDK): kit de desenvolvimento de programa Java. Inclui o JRE
 - Java Standard Edition (Java SE): JRE + JDK
 - Muitas outras versões: Java Card, Java ME Micro-Edition, Java Micro-Profile, Java EE Enterprise-Edition

Linguagem de Programação Java

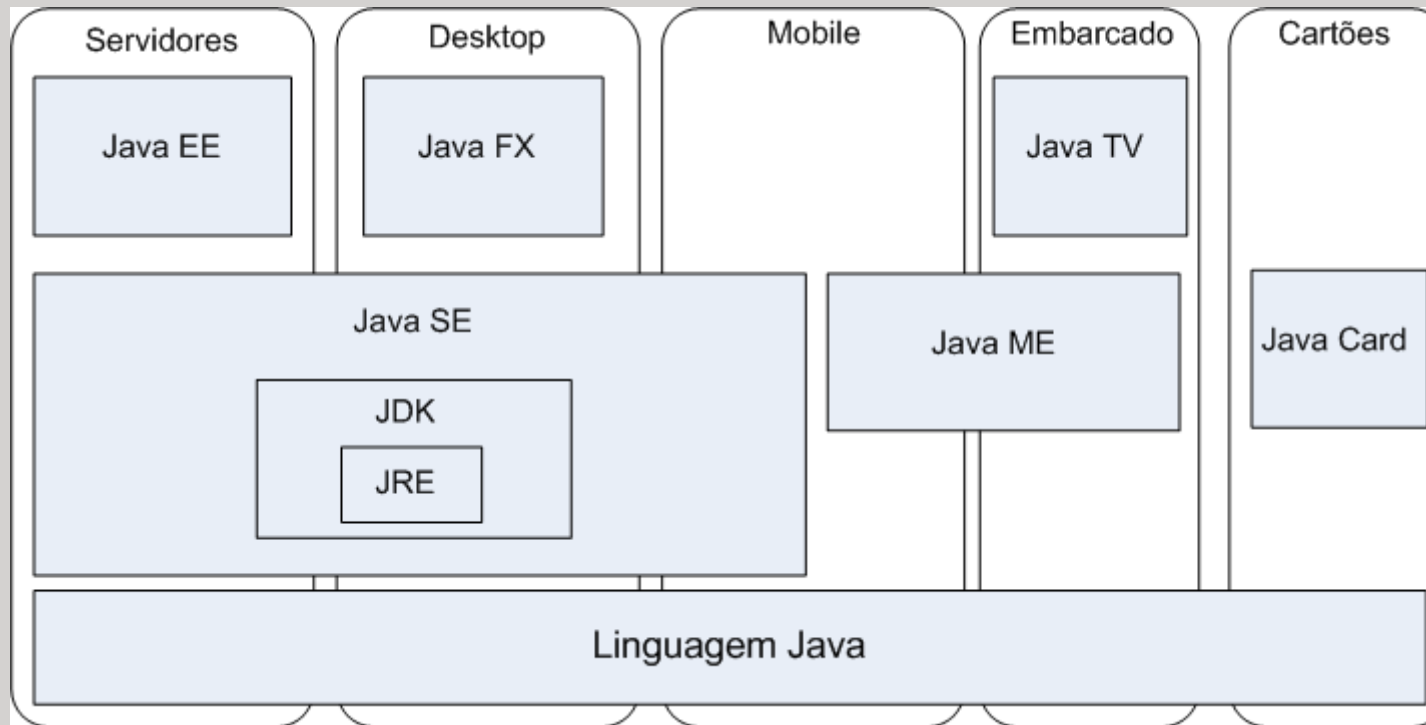
26



- Fonte: <https://dev.java/evolution/>. Acesso em 03/2023.

Linguagem de Programação Java

27



- Figura – Plataformas de Desenvolvimento em Linguagem Java.

Linguagem de Programação Java

28

- Java é multiplataforma: escreva uma vez, execute em qualquer lugar.
- Nenhuma alteração específica de hardware precisa ser incorporada ao código-fonte.
- Código-fonte é compilado em bytecodes. Bytecodes podem ser executados em qualquer computador que possua uma JVM instalada.
- O código compilado (bytecodes) é um arquivo .class
- Programas Java compilados em bytecodes executam em uma Java Virtual Machine (JVM)
- JVM emula o hardware e a CPU do computador.

Linguagem de Programação Java

29

- **Ambiente de Programação Java:**
 - Sistema Operacional
 - Linguagem
 - APIs
 - Bibliotecas de Classe

Linguagem de Programação Java

30

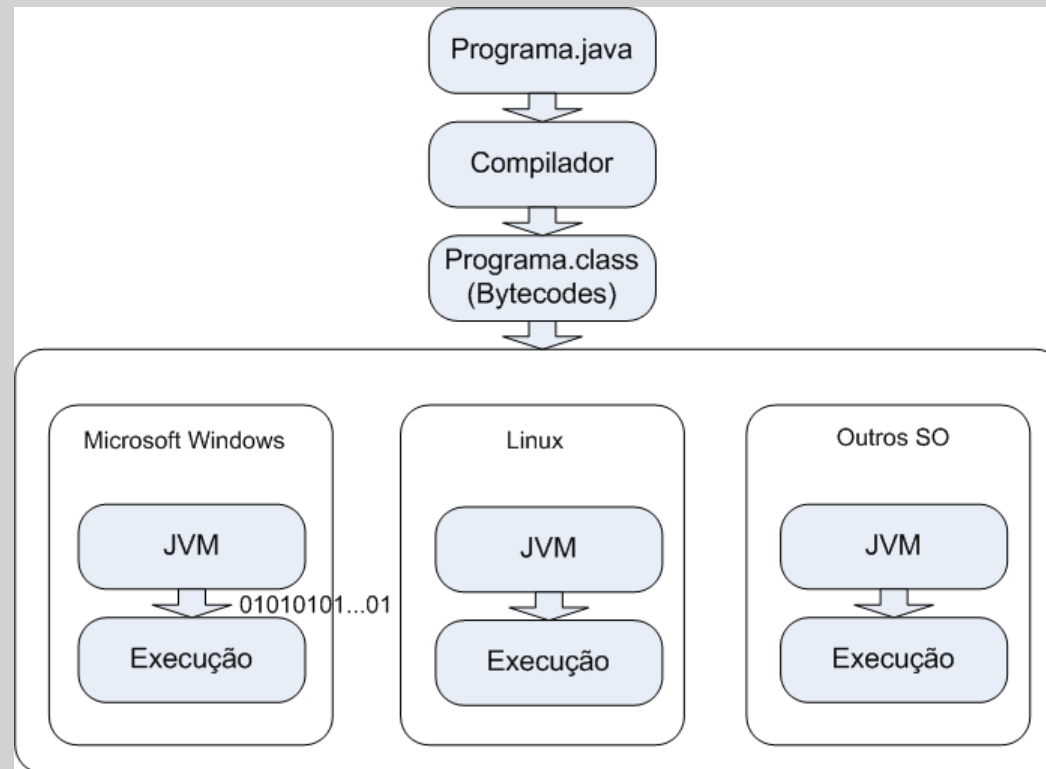


Figura: Fases de Execução de um Programa Java.

Linguagem de Programação Java

31

- **Fases de Execução de um Programa Java:**
 - Edição: programador escreve o programa e salva em disco.
 - Compilação: compilador cria bytecodes
 - Carregamento: armazenagem de bytecodes na memória
 - Verificação: verificação de requisitos de segurança
 - Execução: interpretador traduz bytecodes em código de máquina.

Programação em Java

Programação em Java

33

- Estrutura de um programa Java:
 - Classe: é a unidade básica do código que descreve a lógica do programa.
 - Classe: é um modelo que descreve as propriedades gerais de um objeto.
 - Objeto: é uma instância da classe.
 - Pacote: é um conjunto de classes. Semelhante às bibliotecas da linguagem C.



Revisão

Revisão

35

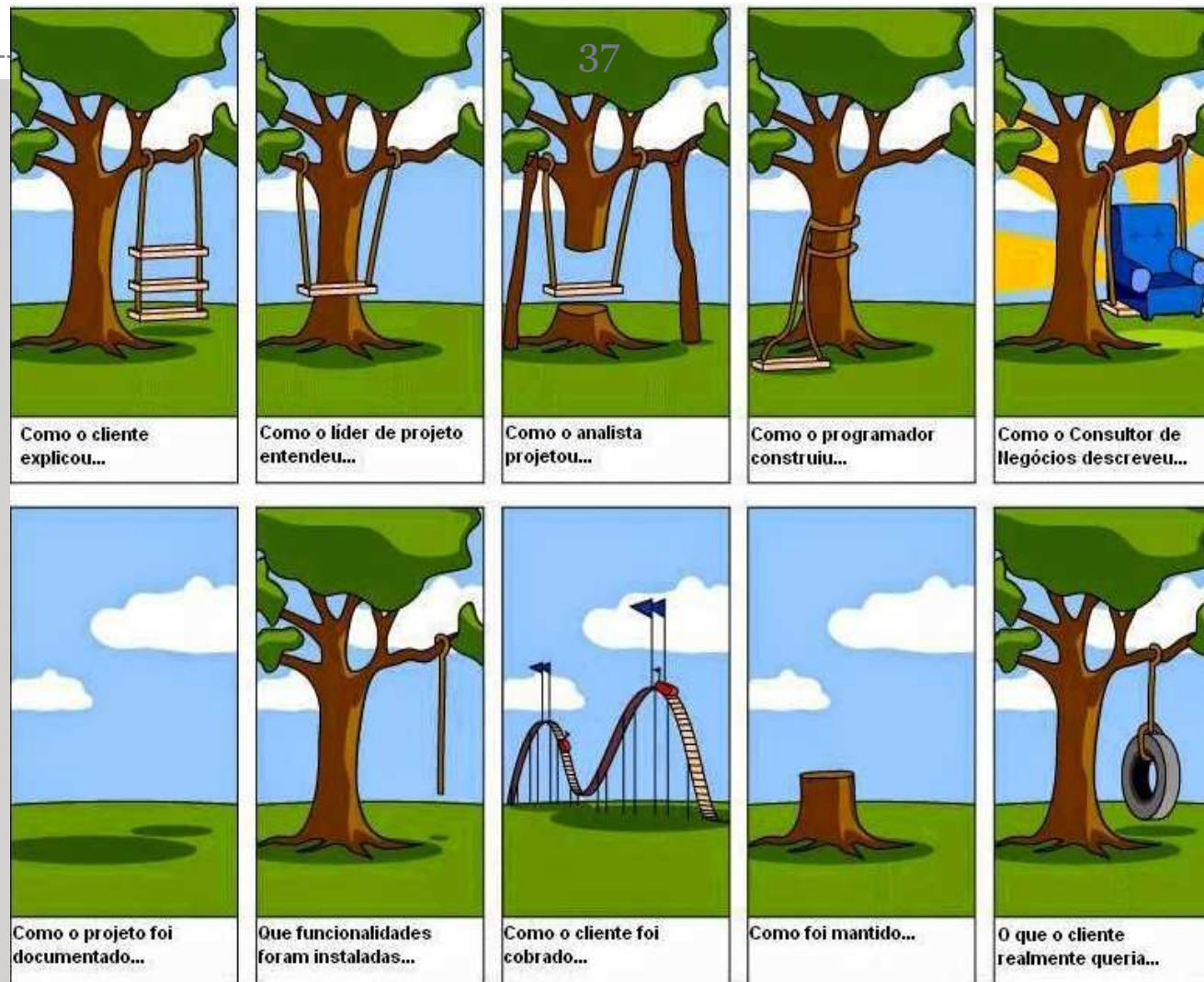
- A Programação Orientada a Objetos (POO) é um paradigma de programação no qual propriedades e comportamentos são armazenados em objetos:
 - Propriedades: são atributos do objeto.
 - Comportamentos: são as ações/operações que o objeto realiza.
 - Classe: é um modelo que descreve objetos.
 - Classe: é um modelo que define as propriedades e comportamentos comuns a todos os objetos.
 - Objeto: é uma instância de uma classe.
 - Objetos: devem ser capazes de interagir uns com os outros através de seus métodos.

Revisão

36

- POO surgiu na década de 70 como uma necessidade do mercado e do ciclo de desenvolvimento do software: redução de complexidade, contribuição de outras áreas (sistemas operacionais, IA, técnicas orientadas a abstração de dados).
- Wegner[8]: POO é um paradigma de programação que utiliza objetos como entidades autônomas que possuem um estado e respondem a mensagens. Classes agrupam objetos que têm atributos e operações em comum. Herança é utilizada para organizar as classes de acordo com características comuns.
- Características da POO:
 - Encapsulamento
 - Ocultamento
 - Fraco acoplamento
 - Modularidade
 - Comunicação via Interfaces
- Java é uma linguagem de Programação Orientada a Objetos.

Para pensar...



Fonte: <http://calvinberschscherer.blogspot.com.br/2014/06/projeto-balanco-no-agile.html>. Acesso em 09/02/2015.

Exercícios

38

<Ver conteúdo na Plataforma de Ensino>



Para lembrar...

39

- Precedência de Operadores

Operador	Significado
()	Parênteses
*, /, %	Multiplicação, Divisão, Módulo
+, -	Adição, Subtração
=	Atribuição

- Avaliação da esquerda para a direita. A cada avaliação, reinicia da esquerda para a direita novamente.
- Expressões delimitadas por Parênteses são avaliadas primeiro. Se há várias expressões, as expressões nos Parênteses mais internos são avaliadas primeiro.
- Multiplicação, divisão e resto têm o mesmo nível de precedência.
- Adição e Subtração têm o mesmo nível de precedência.

Para lembrar...

40

- Precedência de Operadores Relacionais

Operador	Significado
>	X é maior que Y
<	X é menor que Y
>=	X é maior ou igual a Y
<=	X é menor ou igual a Y
==	X é igual a Y
!=	X é diferente de Y

Referências

41

- Referências bibliográficas da disciplina.
- CAPRETZ, L. F. “A Brief History of the Object-Oriented Approach”. In: ACM SIGSOFT Software Engineering Notes, vol.28, no.2, p.6, 2003.
- The Java Tutorials. Disponível em: <https://docs.oracle.com/javase/tutorial>. Acesso em Maio de 2021.