

Programação Orientada a Objetos – (POO0001)

JAVA

Alexandre Mendonça Fava
alexandre.fava@udesc.br

Universidade do Estado de Santa Catarina – UDESC
Programa de Pós-graduação em Computação Aplicada – PPGCA

Programação Orientada a Objetos – (POO0001)



Alexandre Mendonça Fava
alexandre.fava@udesc.br

Universidade do Estado de Santa Catarina – UDESC
Programa de Pós-graduação em Computação Aplicada – PPGCA

Programação Orientada a Objetos – (POO0001)



Alexandre Mendonça Fava
alexandre.fava@udesc.br

Universidade do Estado de Santa Catarina – UDESC
Programa de Pós-graduação em Computação Aplicada – PPGCA

Programação Orientada a Objetos – (POO0001)

Java



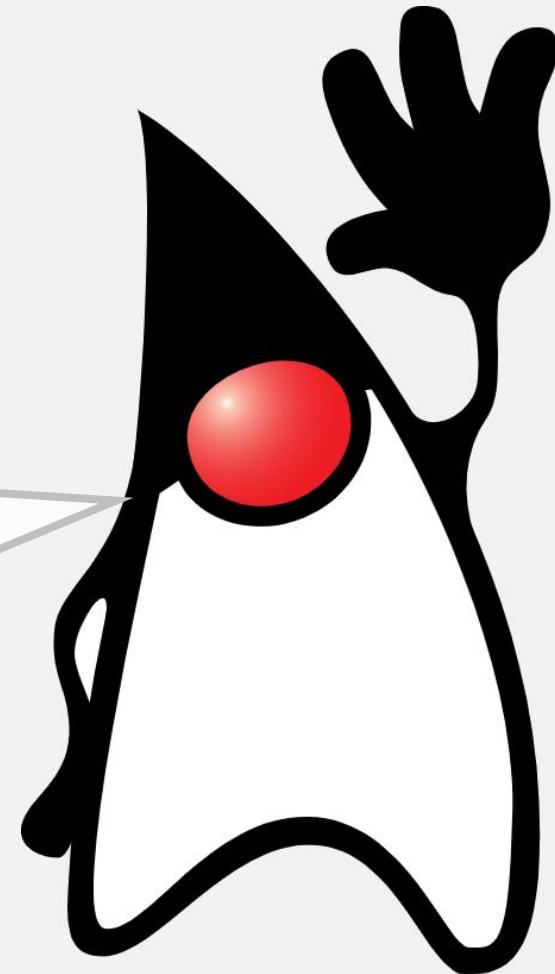
Alexandre Mendonça Fava
alexandre.fava@udesc.br

Universidade do Estado de Santa Catarina – UDESC
Programa de Pós-graduação em Computação Aplicada – PPGCA

Roteiro

- Por que Java?
- Paradigmas de Programação
 - Orientado a Objetos
 - Imperativa
 - Funcional
 - Genérica
- WORA
- Configuração
- Resumo
- Referências

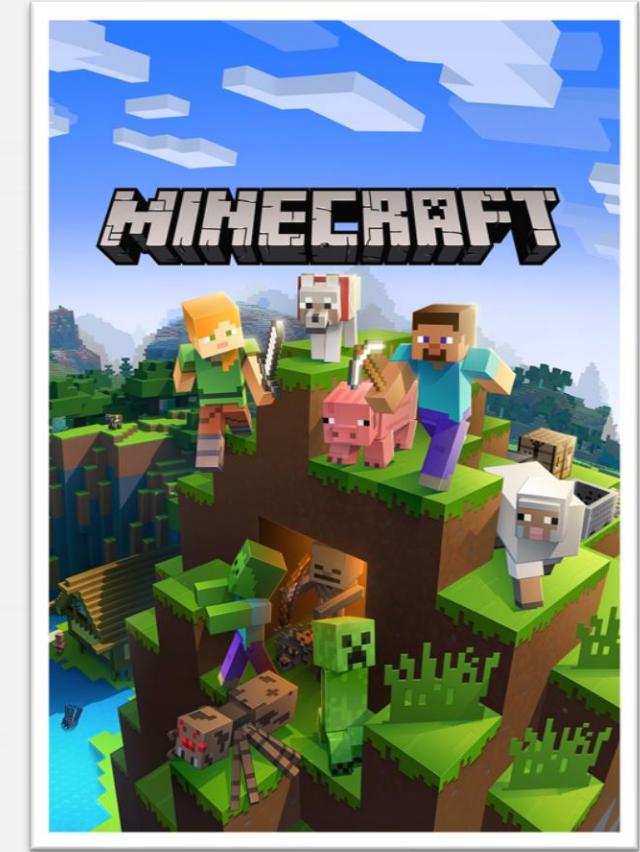
Hora, hora, hora...
Já vamos começar então.



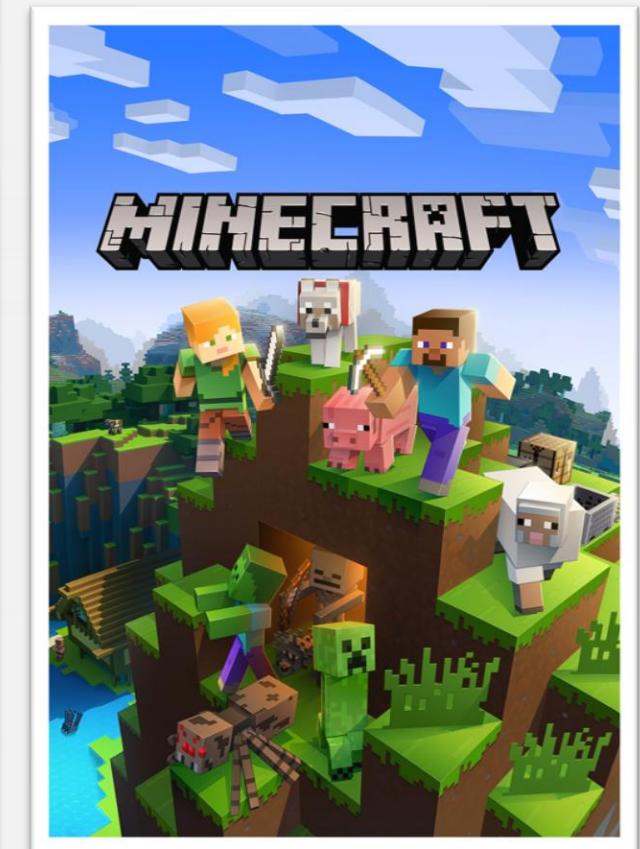
Duke, o Mascote do Java

Por que Java?

Por que Java?



Por que Java?



Por que Java?

DEITEL, Paul. Java: como programar.

Blu-ray Disc™	Caixas automáticos	Canetas inteligentes
Cartões de crédito	Consoles de jogos	Celulares
Cartões inteligentes	Dispositivos médicos	Copiadoras
Decodificadores de TV <i>(set-top boxes)</i>	Estações de pagamento de estacionamento	Desktops (computadores de mesa)
e-Readers	Medidores inteligentes	Impressoras
Eletrodomésticos	Robôs	Interruptores de luz
Imagens por ressonância magnética (MRIs)	Scanners de tomografia computadorizada	Roteadores
Passes de transporte	Sintonizadores de TV a cabo	Sistemas de diagnóstico veicular
Sistemas de aviação	Sistemas de segurança residencial	Sistemas de informação e entretenimento para automóveis
Smartphones	Tablets	Sistemas de navegação GPS
Terminais lotéricos	Termostatos	Televisões

Figura 1.1 | Alguns dispositivos que usam Java.

Por que Java?

<https://www.tiobe.com/tiobe-index/>

Jan 2024	Programming Language	Ratings	Change
1	 Python	13.97%	-2.39%
2	 C	11.44%	-4.81%
3	 C++	9.96%	-2.95%
4	 Java	7.87%	-4.34%
5	 C#	7.16%	+1.43%
6	 JavaScript	2.77%	-0.11%

Por que Java?

<https://stackshare.io/java>

Who uses Java?

COMPANIES

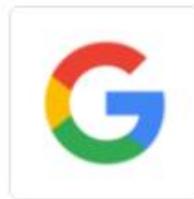
10467 companies reportedly use Java in their tech stacks, including:



Uber



Airbnb



Google



Pinterest



Netflix



Spotify



Amazon



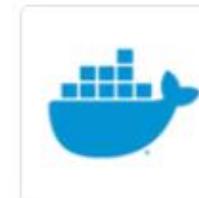
Slack

<https://stackshare.io/java>

Who uses Java?

Java Integrations

Here's a list of all 435 tools that integrate with Java:



Docker



IntelliJ IDEA



Spring Boot



Android SDK



Sentry



Scala



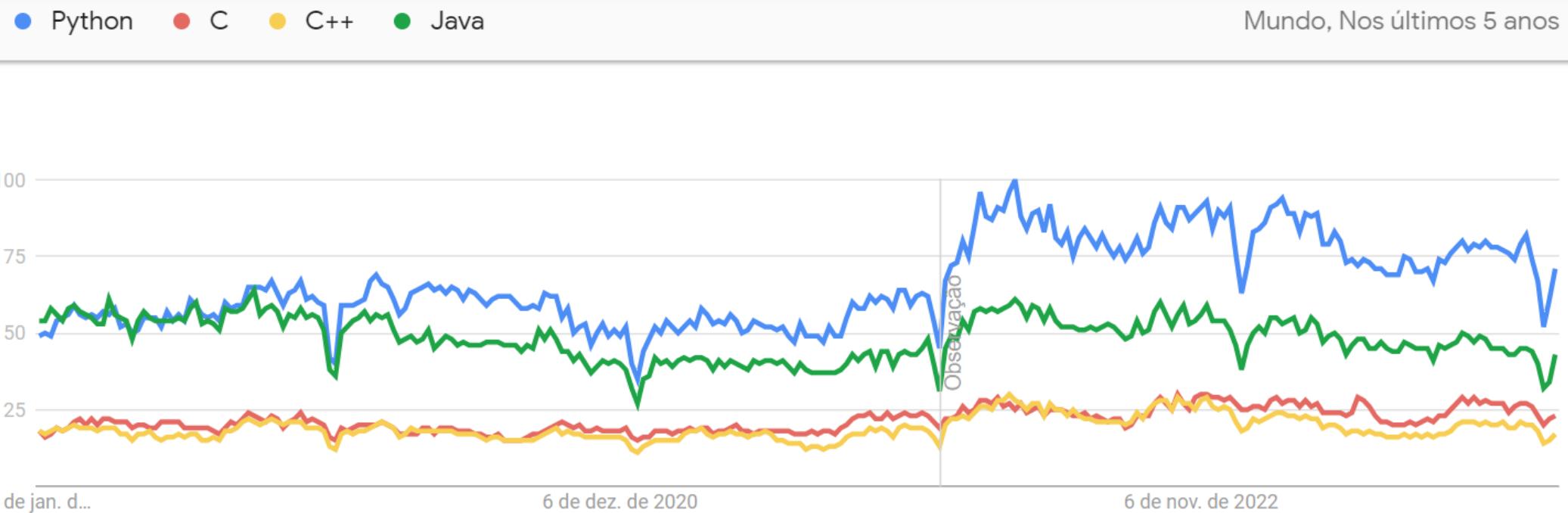
Datadog



JUnit

Por que Java?

<https://trends.google.com.br/trends>



Por que Java?

SCHILD, Herbert. **Java para iniciantes.**

Programação orientada a objetos

A programação orientada a objetos (OOP, object-oriented programming) é a essência de Java. A metodologia orientada a objetos é inseparável da linguagem, e todos os programas Java são, pelo menos até certo ponto, orientados a objetos.

DEITEL, Paul. **Java: como programar.**

Outra razão de Java ser atraente para os cursos universitários é que ela é completamente orientada a objetos. Uma razão que tem aumentado a utilização de C++ tão rapidamente é que ela estende a programação em C para a arena de orientação a objetos. Para a enorme comunidade de programadores C, esta foi uma vantagem poderosa.

Por que Java?

SCHILD, Herbert. Java para iniciantes.

Como você pode observar, a programação em Java é praticamente regrada pelos conceitos de POO, e as classes são a base de qualquer código Java.

Qualquer análise para um novo programa em Java deve partir do entendimento do seu contexto e projeção das classes.

DEITEL, Paul. Java: como programar .

Java é uma linguagem completamente orientada a objetos com forte suporte para técnicas adequadas de engenharia de *software*. É difícil escrever programas semelhantes a C e chamadas procedurais em Java. Você deve criar e manipular objetos. O processamento de erro é embutido na linguagem.

Por que Java?



Por que Java?



<https://pt.wikipedia.org/wiki/C%2B%2B>

≡ C++

文 A 108 línguas ▾

Artigo Discussão

Ler Editar Ver histórico Ferramentas ▾

Origem: Wikipédia, a enciclopédia livre.

C++ (em português: Pronuncia-se "cê mais mais") é uma linguagem de programação compilada multi-paradigma (seu suporte inclui linguagem imperativa, orientada a objetos e genérica) e de uso geral. Desde os anos 1990 é uma das linguagens comerciais mais populares, sendo bastante usada também na academia por seu grande desempenho e base de utilizadores.

Por que Java?



[https://pt.wikipedia.org/wiki/Java_\(linguagem_de_programação\)](https://pt.wikipedia.org/wiki/Java_(linguagem_de_programação))

Java (linguagem de programação) 116 línguas

Artigo Discussão

Ler Editar Ver histórico Ferramentas

Origem: Wikipédia, a encyclopédia livre.

Java é uma linguagem de programação orientada a objetos desenvolvida na década de 90 por uma equipe de programadores chefiada por James Gosling, na empresa Sun Microsystems, que em 2008 foi adquirido pela empresa Oracle Corporation.^[3] Diferente das linguagens de programação modernas, que são compiladas para código nativo, Java é compilada para um bytecode que é interpretado por uma máquina virtual (Java Virtual Machine, abreviada JVM).

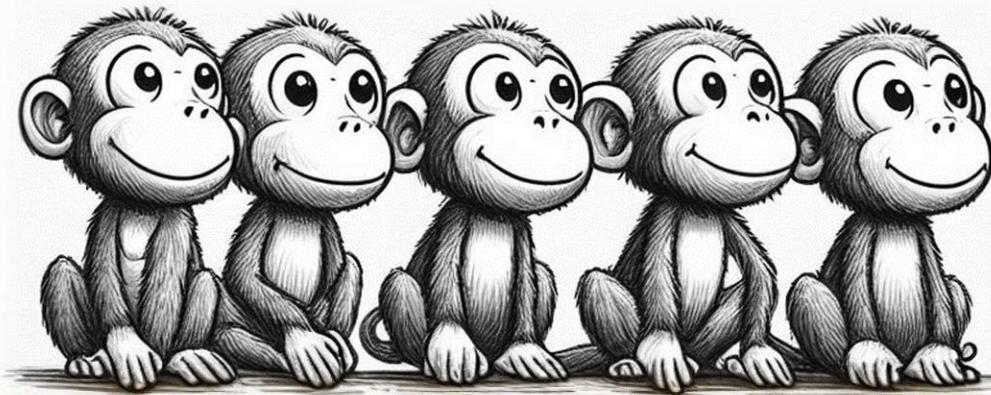
Paradigma

Paradigma



Paradigma

A parábola dos
macacos



Passando giz no taco de sinuca sem saber o motivo



Paradigma

SILVA, Fabricio; LEITE, Márcia; OLIVEIRA, Diego. **Paradigmas de Programação.**

Mas, afinal, o que é um paradigma? Um paradigma representa um padrão de pensamento que guia um conjunto de atividades relacionadas, trata-se de um padrão que define um modelo para a resolução de problemas e regra, basicamente, toda e qualquer linguagem de programação existente.

**É um modelo, um padrão
para a solução de problemas**

Paradigma

Todas as linguagens de programação
querem resolver um problema,
logo

todas as linguagens de programação estão
construídas sobre um paradigma.

Paradigma

Paradigma

(de programação)

É um modelo, um padrão para a solução de problemas

Linguagem

(de programação)

É o vocabulário utilizado para a solução de problemas

Paradigma

Paradigma

(de programação)

É um modelo, um padrão para a solução de problemas

Imperativo

Funcional

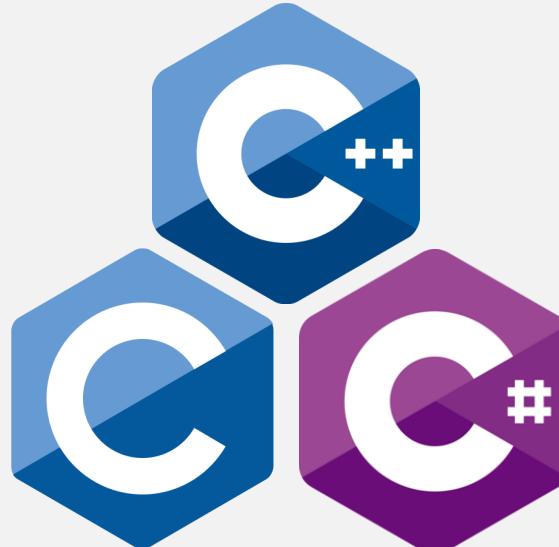
Genérico

Orientado a Objetos

Linguagem

(de programação)

É o vocabulário utilizado para a solução de problemas



Paradigma

Paradigma

(de idioma)

É um modelo, um padrão para a solução de problemas

Fala

Sinais

Tátil

Escrita

Linguagem

(de idioma)

É o vocabulário utilizado para a solução de problemas

Português

Espanhol

LIBRAS

BRAILLE

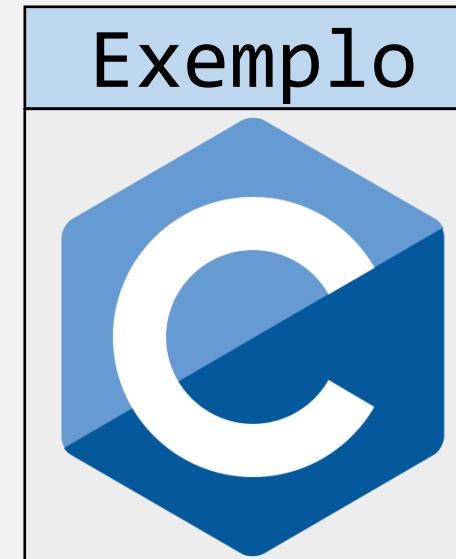
漢字

カタカナ

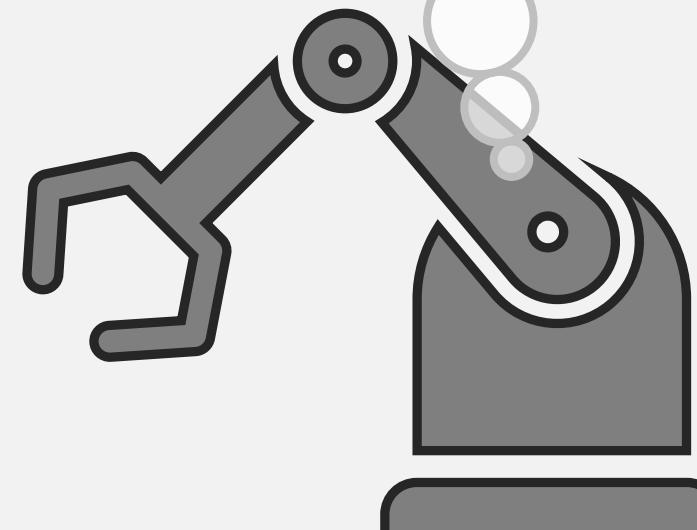
Paradigma

Paradigma: Imperativo

Ei máquina:
faça isso,
depois isso,
depois aquilo.



Humanos e
suas **formas**
imperativas
de dar ordens.



Paradigma: Imperativo

Código

```
1 #include <stdio.h>
2
3 int main() {
4     int i;
5     int lista[9] = {1,2,3,4,5,6,7,8,9};
6     printf("Numeros pares de 1 a 9: ");
7     for(i = 0; i < 9; i++){
8         if(lista[i] % 2 == 0){
9             printf("%d ", lista[i]);
10        }
11    }
12    return 0;
13 }
14 }
```

Paradigma: Funcional

Funções são a base da programação funcional

Uma função pode receber outras funções como argumento



Fique atento

Na programação funcional não existem *loops*. Os laços devem ser criados por meio de recursividade. As funções devem ser puras, ou seja, não podem depender de nenhum elemento externo.

Paradigma: Funcional

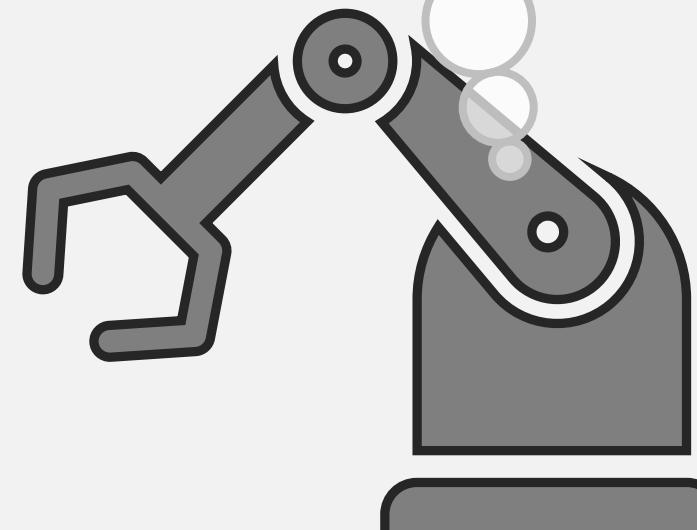
Ei máquina:

$$\lambda x : x + x \rightarrow x[\frac{y}{2}].$$

Exemplo



Humanos e suas **formas matemáticas** de dar ordens.



Paradigma: Funcional

Código

```
1  verificandoPar :: [Integer] -> [Integer]
2  verificandoPar lista = [n | n <- lista, even n]
3
4  main :: IO ()
5  main = do
6    let lista = [1..9]
7    putStrLn $ "Numeros pares de 1 a 9: "
8    print (verificandoPar lista)
9
10
11
12
13
14
```

Paradigma: Genérico

O tipo dos dados de entrada são irrelevantes para a solução

Linguagens de tipagem dinâmicas
meio que já fazem isso

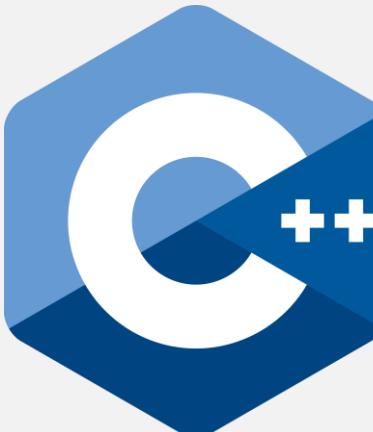
A linguagem de programação **C++** usa **template** como um tipo genérico (ou seja, ele poder ser: **int**, **float**, **char**, **bool**, etc...)

```
T Soma<T>(T p1, T p2) { return p1 + p2; }  
Soma(1, 2);  
Soma(1.5, 2.7);
```

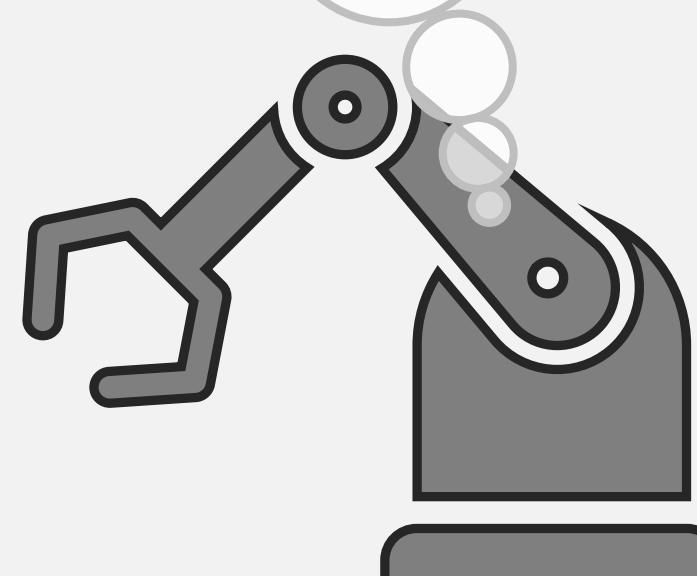
Paradigma: Genérico

Ei máquina:
**faça isso, não
importa a entrada,
dá teu jeito.**

Exemplo



Humanos e
suas **formas
genéricas** de
dar ordens.



Paradigma: Genérico

Código

```
16 int main() {  
17     list<int> numeros = {1, 2, 3, 4, 5, 6, 7, 8, 9};  
18     list<int> pares = filtrarPares(numeros);  
19  
20     cout << "Numeros pares de 1 a 9: ";  
21     for (int numero : pares) {  
22         cout << numero << " ";  
23     }  
24     cout << endl;  
25  
26     return 0;  
27 }  
28 }
```

Paradigma: Genérico

Código

```
1 #include <iostream>
2 #include <list>
3
4 using namespace std;
5 template <typename T>
6 list<T> filtrarPares(list<T> lista){
7     list<T> pares;
8     for (T numero : lista){
9         if (numero % 2 == 0){
10             pares.push_back(numero);
11         }
12     }
13     return pares;
14 }
15 }
```

Paradigma: Orientado a Objetos

Tudo é um objeto, e todo objeto é constituído de atributos e métodos

Os objetos podem interagir uns com os outros



Saiba mais

Por ser algo muito abstrato, a programação orientada a objetos é difícil de aprender. Vários conceitos são artificiais e isso torna o aprendizado bastante complicado.

O resultado que se vê é muito código não orientado ao objeto, mas escrito em linguagens orientadas a objetos.

Paradigma: Orientado a Objetos

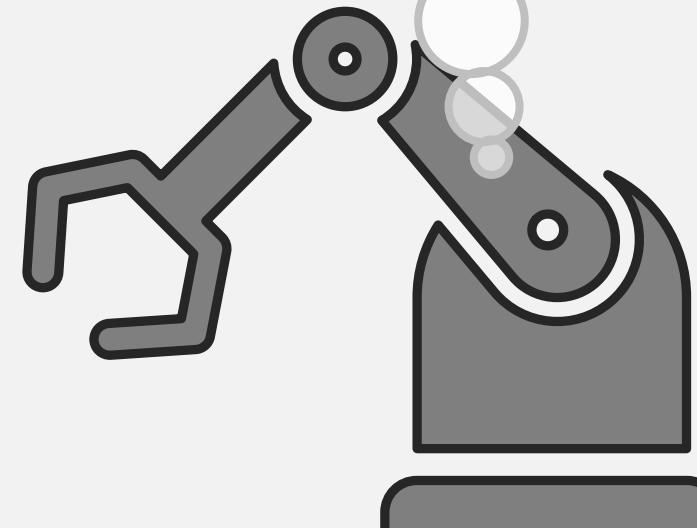
Ei máquina:
**faça isso, com base
naquilo e com um
pouco disso.**



Exemplo



Humanos e
suas **formas de
reaproveitar
ordens.**



Paradigma: Orientado a Objetos

Ei máquina:
**faça isso, com base
naquilo e com um
pouco disso.**

Exemplo



Humanos e
suas **formas de
reaproveitar
ordens.**

DEITEL, Paul. Java: como programar.

Antes do Java SE 8, a linguagem suportava três paradigmas de programação —
programação imperativa, *programação orientada a objetos* e *programação genérica*.

O Java SE 8 acrescenta a *programação funcional*.

Paradigma: Orientado a Objetos

Polimorfismo

Herança

Composição

Encapsulamento

Paradigma: Orientado a Objetos

Polimorfismo



Herança



Composição



Encapsulamento



Abstração

Paradigma: Orientado a Objetos

Código (Main.java)

```
1 import java.util.ArrayList;
2 import java.util.List;
3
4 public class Main {
5     public static void main(String[] args) {
6         List<Integer> numeros = new ArrayList<>();
7         for (int i = 1; i <= 9; i++) {
8             numeros.add(i);
9         }
10        Lista lista = new Lista(numeros);
11        lista.imprimirNumerosPares();
12    }
13}
14
```

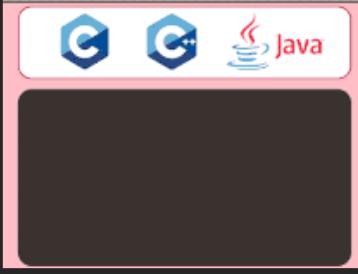
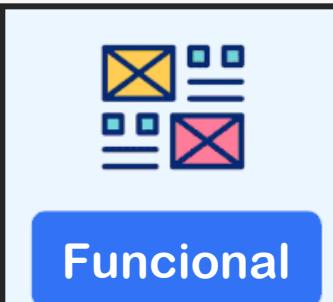
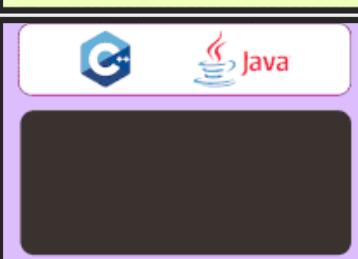
Paradigma: Orientado a Objetos

Código (Lista.java)

```
1 public class Lista{  
2     private List<Integer> numeros;  
3     public Lista(List<Integer> numeros){  
4         this.numeros = numeros;  
5     }  
6     public void imprimirNumerosPares(){  
7         System.out.print("Numeros pares de 1 a 9: ");  
8         for(Integer numero : numeros){  
9             if (numero % 2 == 0){  
10                 System.out.print(numero + " ");  
11             }  
12         }  
13     }  
14 }
```

Paradigmas de Programação

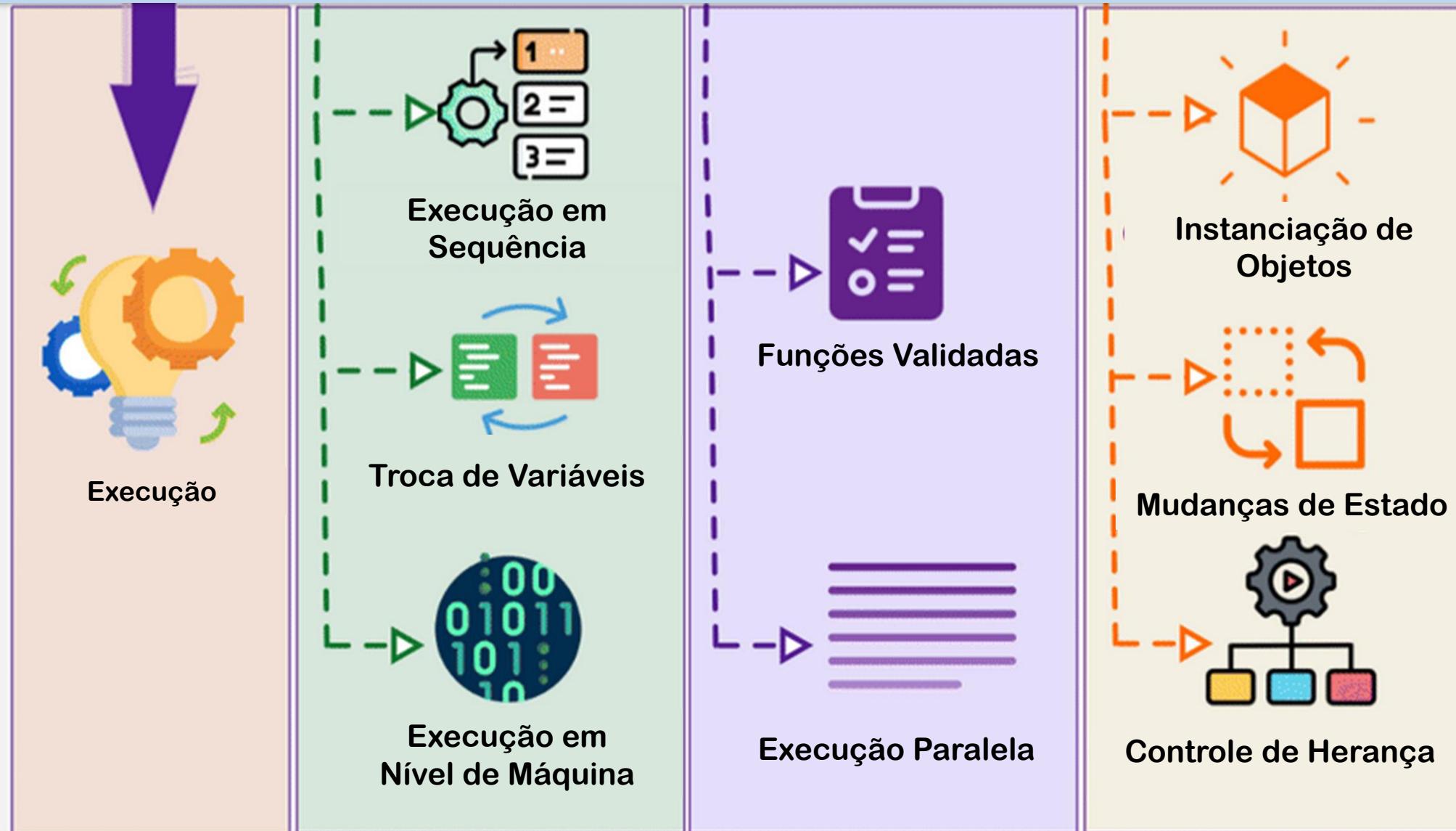
Paradigmas de Programação

		Define uma sequência de passos para a solução de um problema
		Define objetos em termos de atributos e métodos
		Define uma função matemática para a solução de um problema
		Define tipos de dados genéricos que independem da entrada

Paradigmas de Programação



Paradigmas de Programação



WORA

Write Once, Run Anywhere

Escreva uma vez, execute em qualquer lugar

Execução

Execução

Compiler	Author	Microsoft Windows	Unix-like	Other OSs	Bare machine	License type
Acorn C/C++	Acorn and Codemist	No	No	RISC OS	?	Proprietary
AMD Optimizing C/C++ Compiler (AOCC)	AMD	No	Yes	No	?	Proprietary
Aztec C	Manx Software Systems	No	No	CP/M, CP/M-86, DOS, Classic Mac OS	?	Proprietary
Amsterdam Compiler Kit	Andrew Tanenbaum and Ceriel Jacobs	No	Yes	Yes	?	BSD
BDS C	BD Software	No	No	CP/M	?	Public domain

Execução

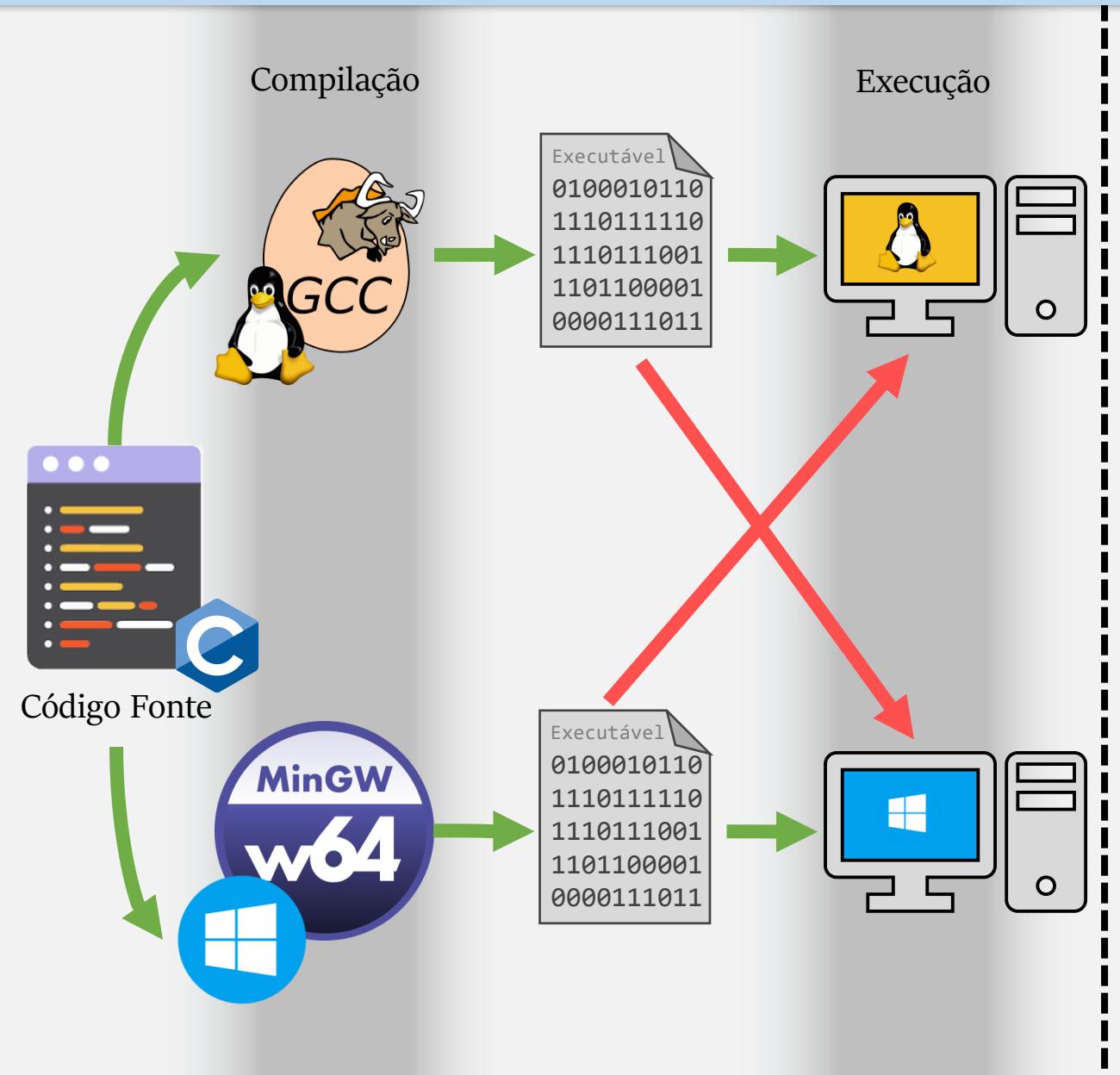
Edison Design Group	Edison Design Group	Yes	Yes	Yes	?	Proprietary
GCC C	GNU Project	MinGW, Cygwin, WSL	Yes	IBM mainframe, AmigaOS, VMS, RTEMS, DOS ^[11]	Yes	GPL
IAR C/C++ Compilers	IAR Systems	Yes	No	No	?	Proprietary
Intel oneAPI DPC++/C++ Compiler (icx)	Intel	Yes	Linux	No	?	Freeware (optional priority support)
Intel C++ Compiler Classic (icc)	Intel	Yes	Linux, macOS	No	?	Freeware (optional priority support)

Execução



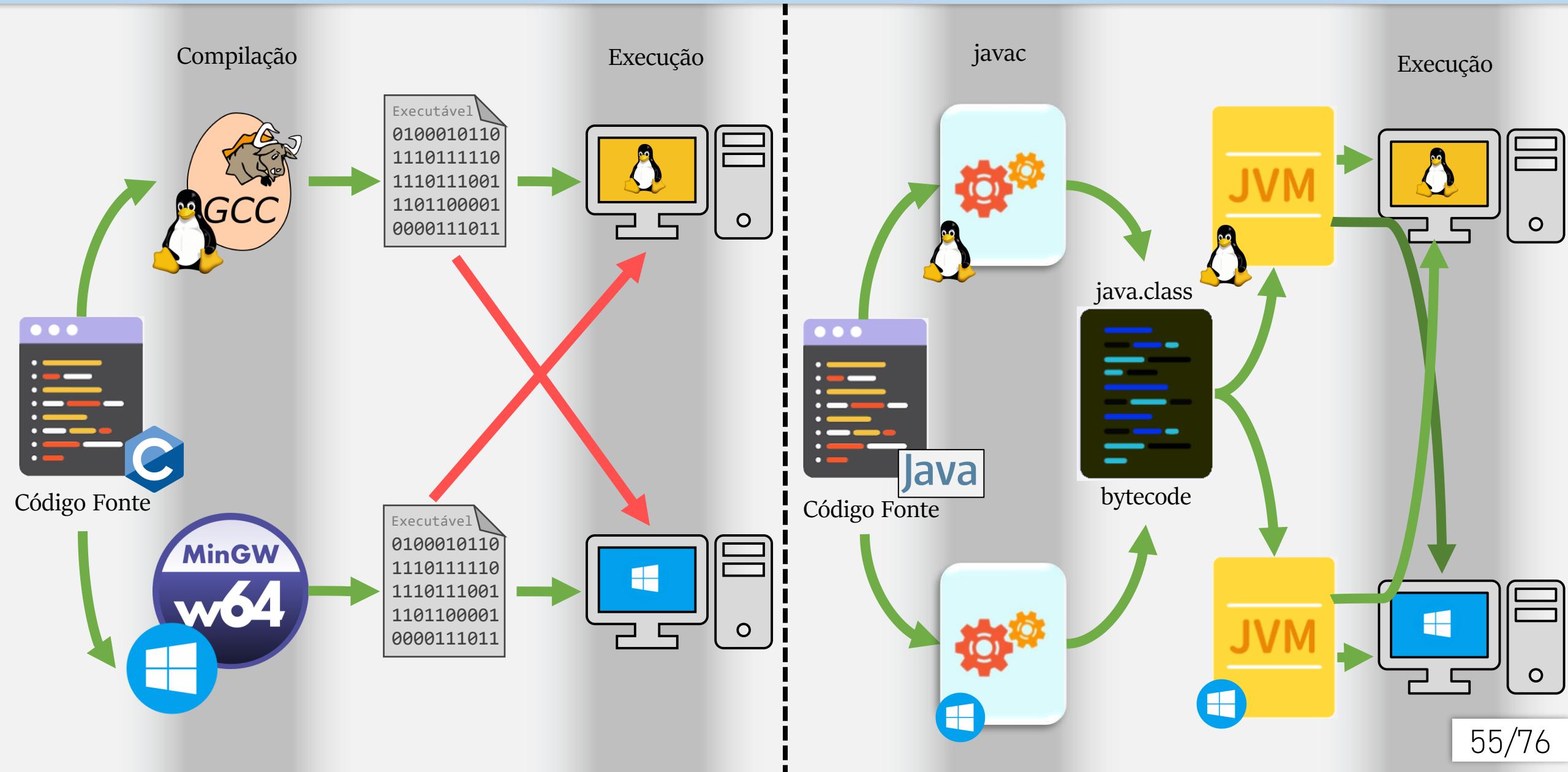
Java

Execução



Java

Execução

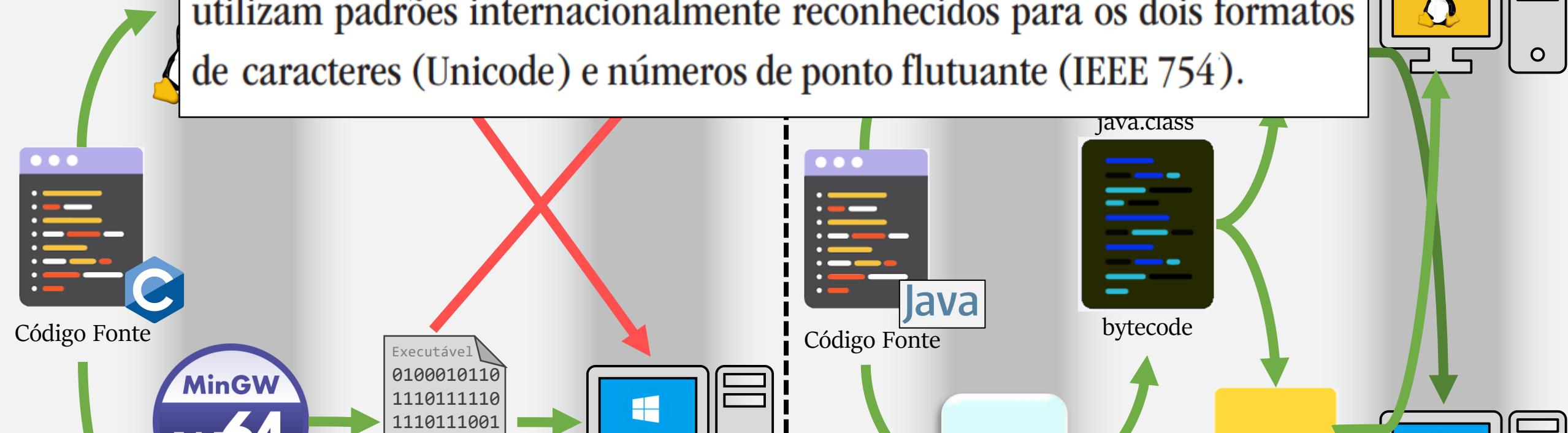


Execução

DEITEL, Paul. Java: como programar.

Execução

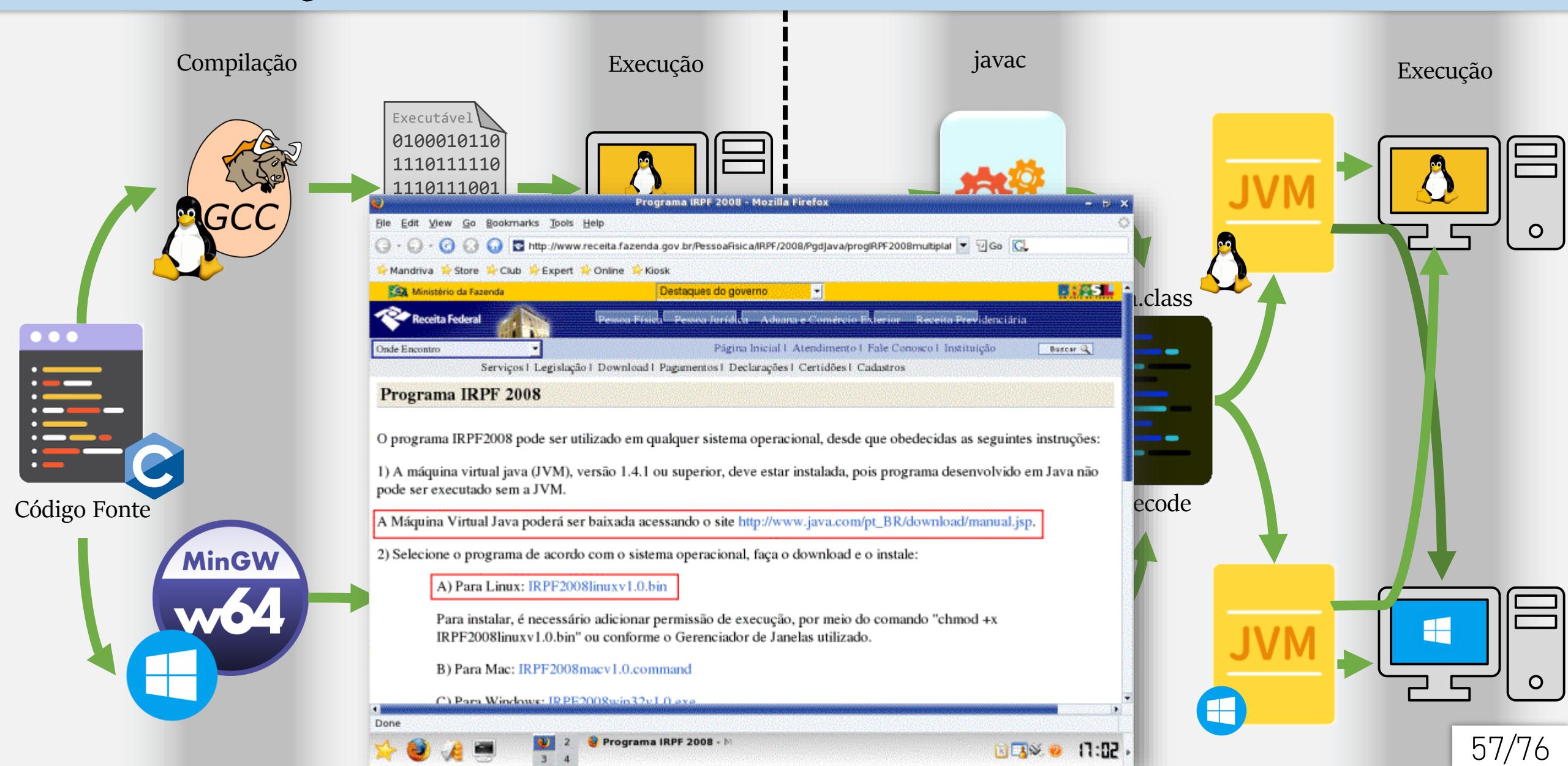
Como os projetistas do Java querem assegurar a portabilidade, eles utilizam padrões internacionalmente reconhecidos para os dois formatos de caracteres (Unicode) e números de ponto flutuante (IEEE 754).



DEITEL, Paul. Java: como programar.

Os programas Java normalmente passam por cinco fases para serem executados (Fig. 1.1). São as seguintes: *edição, compilação, carga, verificação e execução*. Discutimos estes conceitos no contexto do Java 2 Software De

Execução



Execução

SCHILDT, Herbert. Java para iniciantes.

P: Por que Java usa Unicode?

R:

Execução

SILVA, Fabricio; LEITE, Márcia; OLIVEIRA, Diego. **Paradigmas de Programação.**

De fato, a característica mais marcante da linguagem de programação Java está relacionada a sua portabilidade, pois os sistemas construídos não são compilados em código nativo da plataforma. Programas em Java são compilados para um *bytecode*, que é executado por uma máquina virtual, a Java *virtual machine*, que permite que os programas escritos em Java possam ser rodados em qualquer plataforma compatível com a sua máquina virtual.

Execução

SCHILDT, Herbert. Java para iniciantes.

Converter um programa Java em bytecode facilita muito a execução de um programa em uma grande variedade de ambientes, porque só a JVM tem que ser implementada para cada plataforma. Uma vez que o pacote de tempo de execução estiver presente em um determinado sistema, qualquer programa Java poderá ser executado nele. Lembre-se, embora os detalhes da JVM sejam diferentes de uma plataforma para outra, todas entendem o mesmo bytecode Java.

Execução

SCHILDT, Herbert. Java para iniciantes.

O fato de um programa Java ser executado pela JVM também ajuda a torná-lo seguro. Já que a JVM está no controle, ela pode reter o programa e impedi-lo de gerar efeitos colaterais fora do sistema. A segurança também é aumentada por certas restrições existentes na linguagem Java.

SCHILDT, Herbert. Java para iniciantes.

Quando um programa é interpretado, geralmente ele é executado de modo mais lento do que o mesmo programa sendo executado quando compilado para código executável. No entanto, em Java, a diferença entre os dois não é tão grande. Já que o bytecode foi altamente otimizado, seu uso permite que a JVM execute programas de maneira mais rápida do que o esperado.

Execução

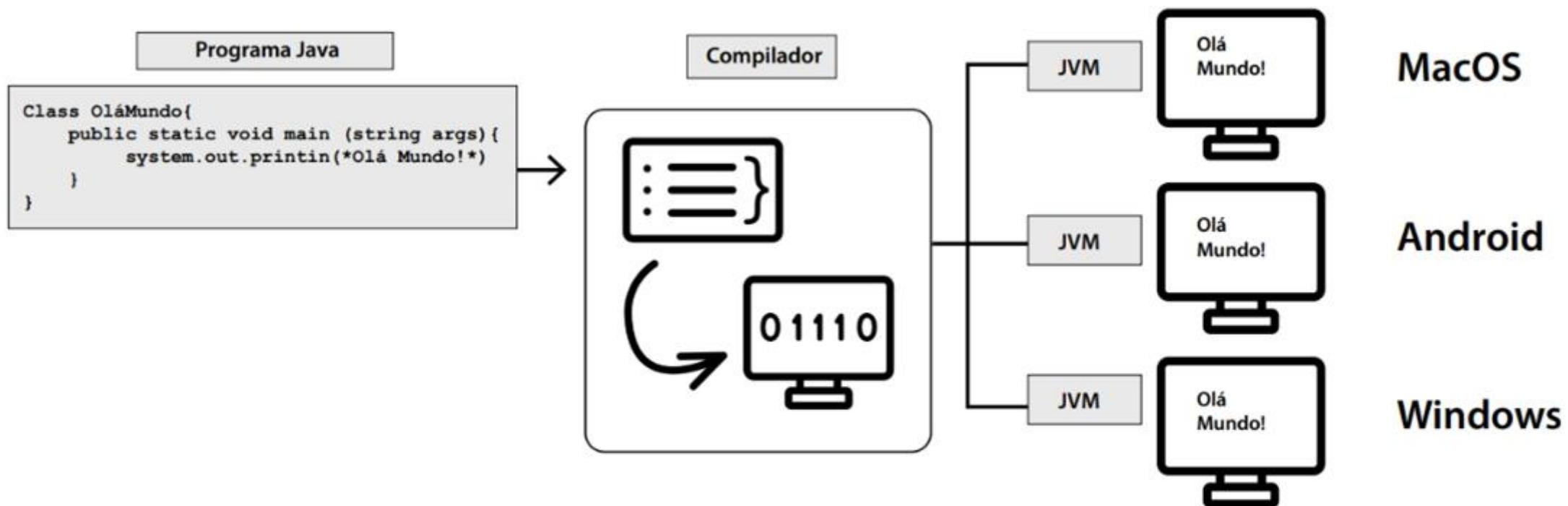
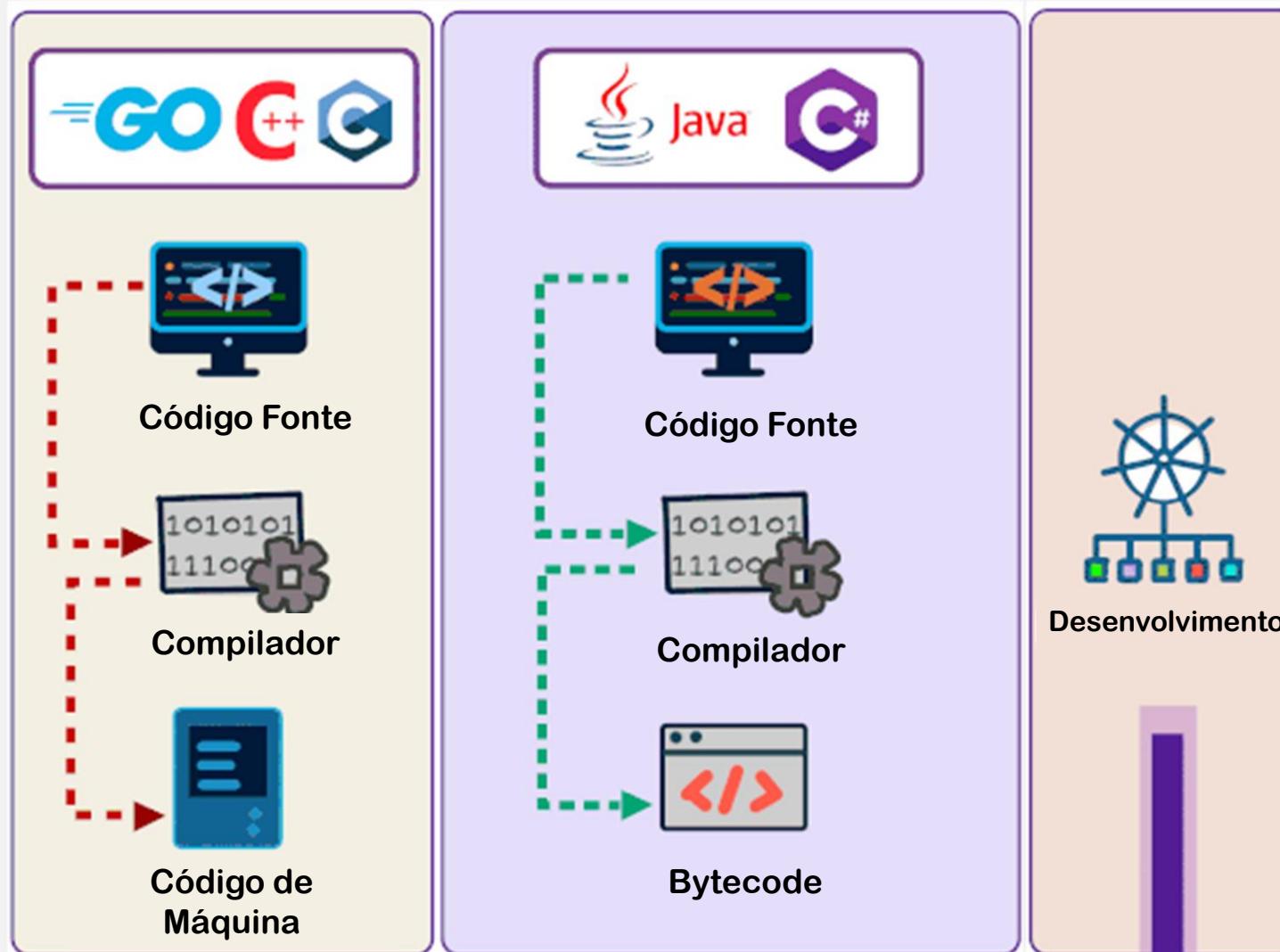


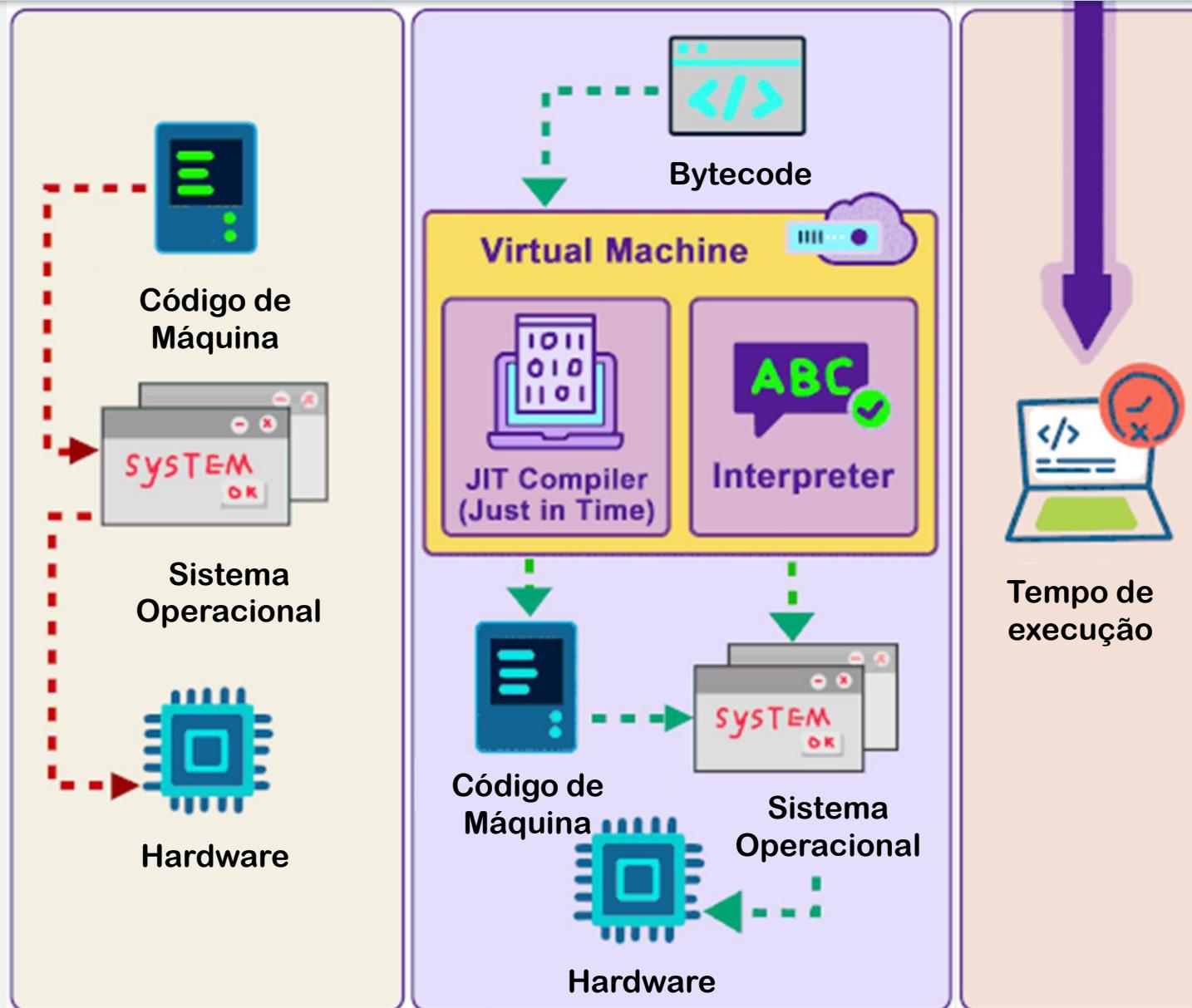
Figura 2. JVM.

Fonte: Adaptada de DreamStockIcons/Shutterstock.com.

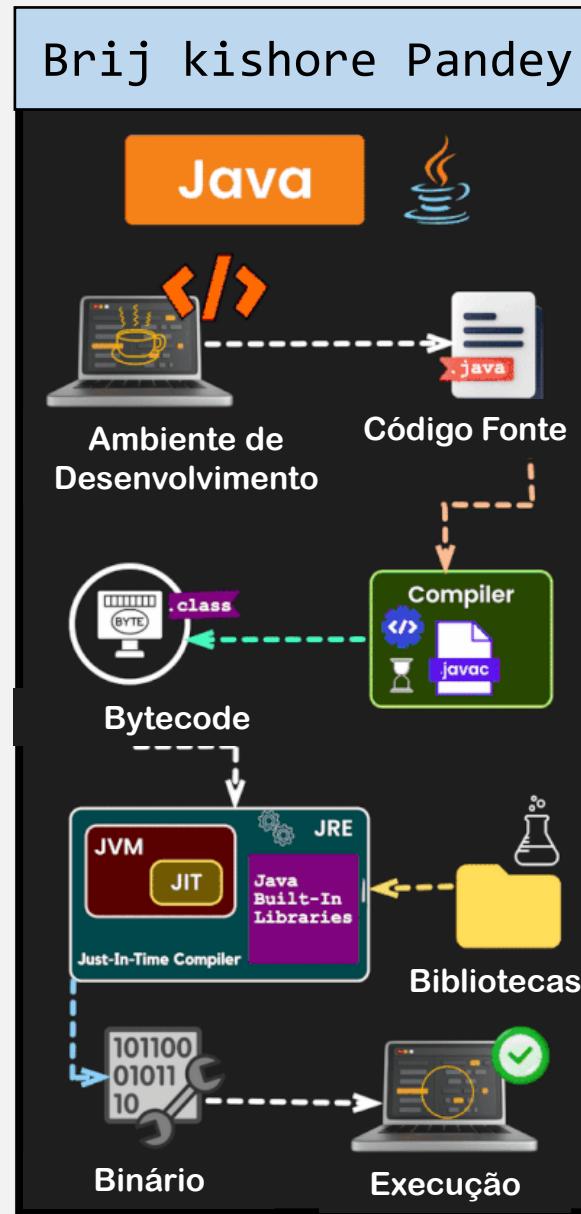
Execução



Execução



Execução



Configuração

<https://www.oracle.com/br/java/technologies/downloads/#jdk21-windows>

Java downloads Tools and resources Java archive

JDK Development Kit 21.0.1 downloads

JDK 21 binaries are free to use in production and free to redistribute, at no cost, under the [Oracle No-Fee Terms and Conditions \(NFTC\)](#).

JDK 21 will receive updates under the NFTC, until September 2026, a year after the release of the next LTS. Subsequent JDK 21 updates will be licensed under the [Java SE OTN License \(OTN\)](#) and production use beyond the [limited free grants](#) of the OTN license will [require a fee](#).

Linux macOS **Windows**

Product/file description	File size	Download
x64 Compressed Archive	185.39 MB	https://download.oracle.com/java/21/latest/jdk-21_windows-x64_bin.zip (sha256)
x64 Installer	163.82 MB	https://download.oracle.com/java/21/latest/jdk-21_windows-x64_bin.exe (sha256)
x64 MSI Installer	162.60 MB	https://download.oracle.com/java/21/latest/jdk-21_windows-x64_bin.msi (sha256)

Configuração

Variáveis de Ambiente

Variáveis de usuário para Windows

Variável	Valor
OneDrive	C:\Users\Windows\OneDrive
Path	C:\Users\Windows\AppData\Local\Microsoft\
TEMP	C:\Users\Windows\AppData\Local\Temp
TMP	C:\Users\Windows\AppData\Local\Temp

[Novo...](#) [Editar...](#) [Excluir](#)

Variáveis do sistema

Variável	Valor
asl.log	Destination=file
ComSpec	C:\WINDOWS\system32\cmd.exe
DriverData	C:\Windows\System32\Drivers\DriverData
JAVA_HOME	C:\Program Files\Android\Android Studio\jre
NUMBER_OF_PROCESSORS	12
OS	Windows_NT
Path	C:\Program Files\Common Files\Oracle\Java\javapath;C:\Program ...

[Novo...](#) [Editar...](#) [Excluir](#)

[OK](#) [Cancelar](#)

https://www.

Java downloads Tools and re

JDK Development Kit 21

JDK 21 binaries are free to use in

JDK 21 will receive updates under
under the **Java SE OTN License (NFTC)**.

Linux macOS Windows

Product/file description

x64 Compressed Archive

x64 Installer

x64 MSI Installer

dk21-windows

s-x64_bin.zip (sha256)

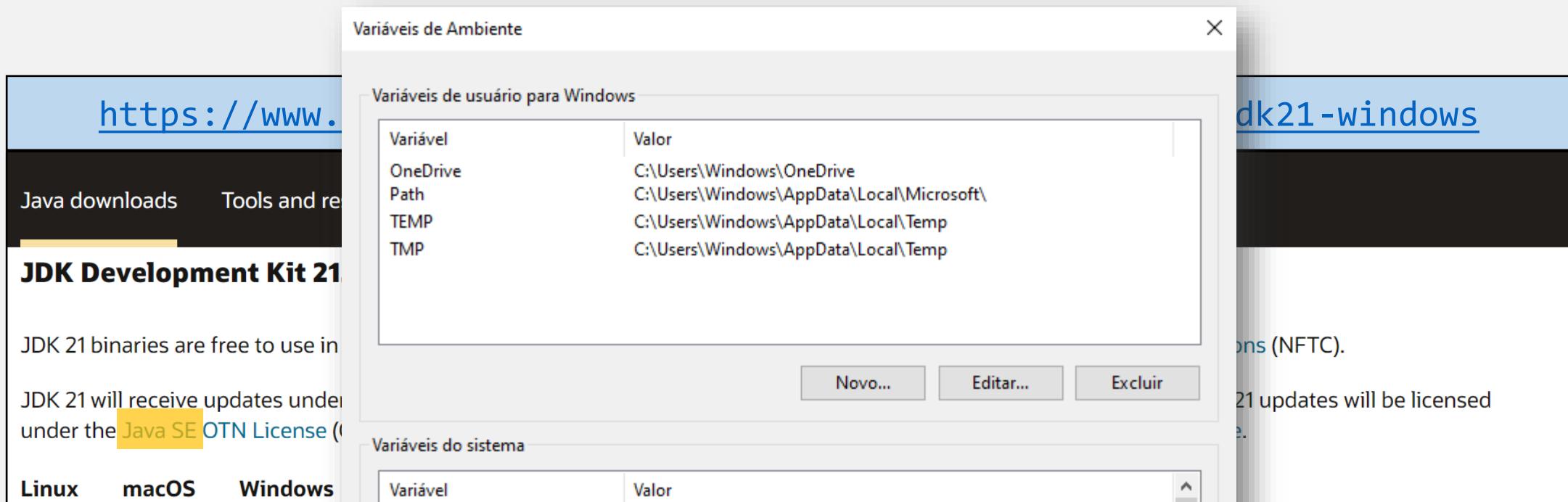
s-x64_bin.exe (sha256)

s-x64_bin.msi (sha256)

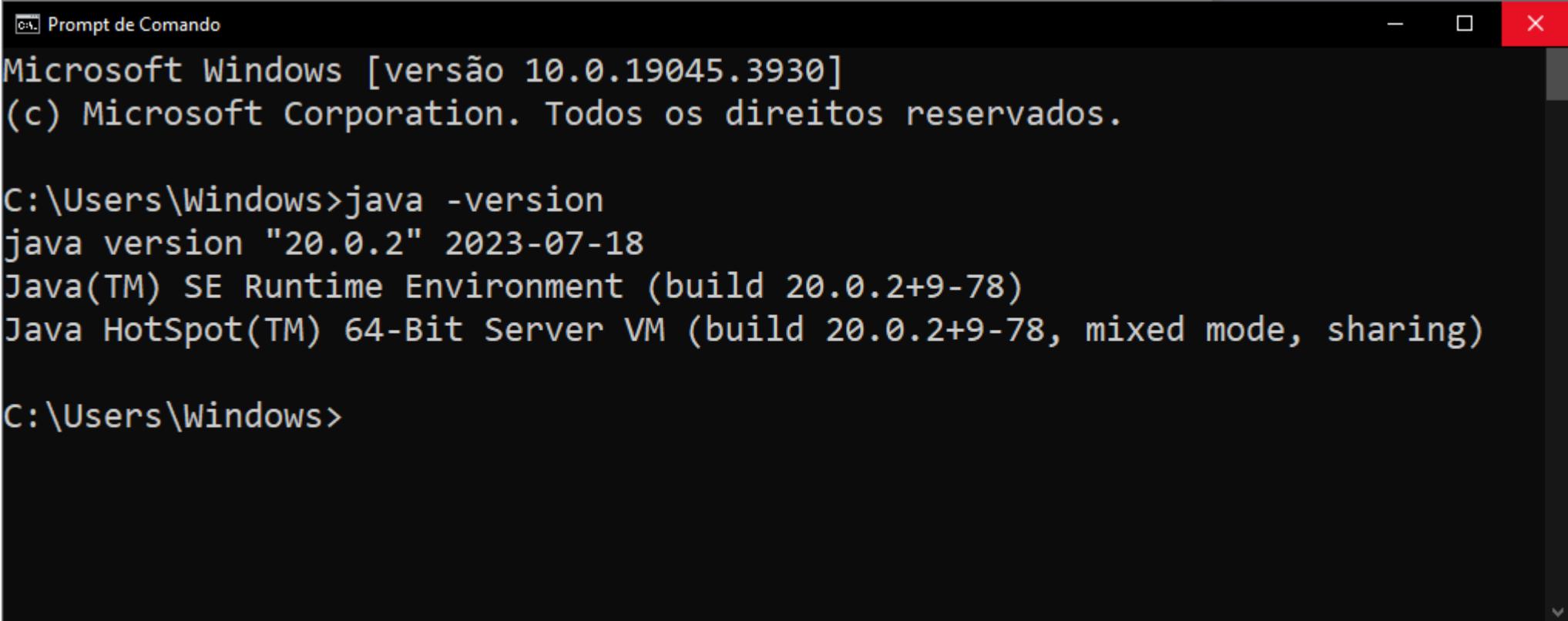
Configuração

SCHILD'T, Herbert. Java para iniciantes.

Mais uma coisa: o JDK é executado no ambiente de prompt de comando e usa ferramentas de linha de comando. Ele não é um aplicativo de janelas. Também não é um ambiente de desenvolvimento integrado (IDE – integrated development environment).



Configuração



A screenshot of a Windows Command Prompt window titled "Prompt de Comando". The window shows the following text:

```
Microsoft Windows [versão 10.0.19045.3930]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Users\Windows>java -version
java version "20.0.2" 2023-07-18
Java(TM) SE Runtime Environment (build 20.0.2+9-78)
Java HotSpot(TM) 64-Bit Server VM (build 20.0.2+9-78, mixed mode, sharing)

C:\Users\Windows>
```

SCHILD, Herbert. Java para iniciantes.

Mais uma coisa: o JDK é executado no ambiente de prompt de comando e usa ferramentas de linha de comando. Ele não é um aplicativo de janelas. Também não é um ambiente de desenvolvimento integrado (IDE – integrated development environment).

Configuração

The image shows two windows side-by-side. The left window is a Notepad titled "Programa.java - Bloco de Notas" containing the following Java code:

```
public class Programa{
    public static void main(String[] args){
        System.out.println("Olá Mundo");
    }
}
```

The right window is a Command Prompt titled "Prompt de Comando" showing the output of running the Java program:

```
Microsoft Windows [versão 10.0.19045.3930]
(c) Microsoft Corporation. Todos os direitos reservados.

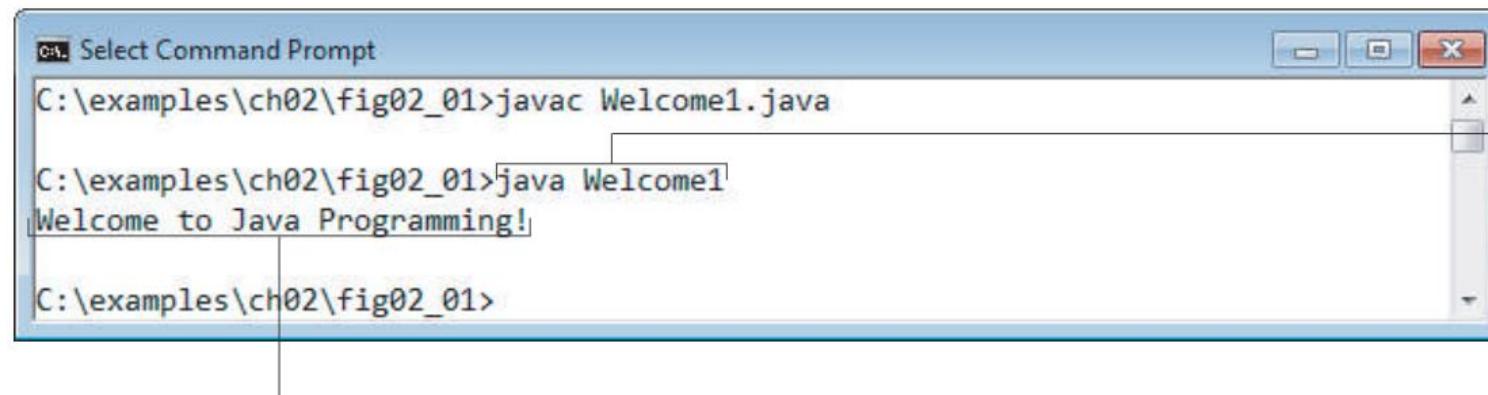
C:\Users\Windows\Desktop>javac Programa.java

C:\Users\Windows\Desktop>java Programa
Olá Mundo

C:\Users\Windows\Desktop>
```

Configuração

DEITEL, Paul. Java: como programar.



The screenshot shows a Windows Command Prompt window titled "Select Command Prompt". The command line shows the user navigating to the directory "C:\examples\ch02\fig02_01" and running the command "javac Welcome1.java". The output of the command shows the Java code "Welcome to Java Programming!". A mouse cursor is visible over the command line. The window has standard Windows-style buttons for minimize, maximize, and close.

```
C:\examples\ch02\fig02_01>javac Welcome1.java
C:\examples\ch02\fig02_01>java Welcome1
Welcome to Java Programming!
C:\examples\ch02\fig02_01>
```

Você digita esse comando para executar o aplicativo

O programa gera a saída na tela
Welcome to Java Programming !

Figura 2.2 | Executando Welcome1 do Command Prompt.

Configuração

<https://moodle.joinville.udesc.br/>

 JOINVILLE Página inicial Painel Meus cursos  

 Departamento de Ciência da Computação / Bacharelado em Ciência da Computação / 2024/1 CC / 2024/1_CCT_CCI192-03B_POO0001 / Exercício 0

 **Exercício 0**

[Marcar como feito](#)

 Descrição  Enviar  Editar  Visualizar envios

 **Data de entrega:** sábado, 9 mar. 2024, 17:00
 **Número máximo de arquivos:** 10
Tipo de trabalho:  Trabalho individual

Criar um programa simples em Java que exiba a mensagem "Olá, Mundo!".

72/76

Configuração

<https://moodle.joinville.udesc.br/>

The screenshot shows a Moodle course page for the '2024/1_CCT_CCI192-03B_POO0001' course. A file upload dialog box is open, prompting the user to 'Criar um novo arquivo' (Create a new file). The file name is set to 'Nome.java'. The dialog has 'OK' and 'Cancelar' buttons. In the background, there's a navigation bar with the UDESC logo, menu icons, and links for 'Página inicial', 'Painel', and 'Meus cursos'. There are also notification icons for messages and alerts.

Departamento de Ciência da Computação / Bacharelado em Ciência da Computação / 2024/1 CC
/ 2024/1_CCT_CCI192-03B_POO0001 /

VPL

Descrição Enviar

Nome do novo arquivo Nome.java

OK Cancelar

73/76

Configuração

<https://moodle.joinville.udesc.br/>

UDESC | JOINVILLE Página inicial Painel Meus cursos ✉ ...

Departamento de Ciência da Computação / Bacharelado em Ciência da Computação / 2024/1 CC
/ 2024/1_CCT_CCI192-03B_POO0001 /

VPL

Descrição Enviar Editar Visualizar envios

Nome.java

1 |

Ln 1, Col 1 Java

74/76

Configuração

<https://netbeans.apache.org/front/main/download/nb20/>



Apache NetBeans

Search the docs

Community

Downloading Apache NetBeans 20

Apache NetBeans 20 was released on December 1, 2023.

Apache NetBeans 20 is available for download from your closest Apache mirror.

Binaries (Platform Independent):

- netbeans-20-bin.zip (SHA-512, PGP ASC)

Installers and Packages:

- Apache-NetBeans-20r1-bin-windows-x64.exe (SHA-512, PGP ASC)
- Apache-NetBeans-20.pkg (SHA-512, PGP ASC)
- apache-netbeans_20-1_all.deb (SHA-512, PGP ASC)
- apache-netbeans-20-0.noarch.rpm (SHA-512, PGP ASC)
- Linux snap package

Resumo

- **Paradigma de Programação:** Define um modelo para solução de problemas.
- **Programação Orientada a Objetos:** O paradigma da orientação a objetos é inseparável da linguagem Java, embora possa não ser muito perceptível em programas pequenos.
- **Ambiente de Desenvolvimento:** Quando instalado e configurado, os códigos em Java podem ser escritos em softwares mais simples como **blocos de notas**, até softwares mais avançados como **NetBeans**.
- **Independência de plataforma:** O código Java é compilado para *bytecode*, que é executado pela Máquina Virtual Java (**JVM**).
- **Orientação a Objetos:** Não é apenas uma linguagem de programação, mas também uma filosofia de desenvolvimento de software (e.g. **UML**).

Referências

SCHILDT, Herbert. **Java para iniciantes: crie, compile e execute.** 6. ed. Porto Alegre: Bookman, 2015. Disponível em: <<https://app.minhabiblioteca.com.br/reader/books/9788582603376>>.

SILVA, Fabricio; LEITE, Márcia; OLIVEIRA, Diego. **Paradigmas de Programação.** Porto Alegre: SAGAH, 2019. Disponível em: <<https://app.minhabiblioteca.com.br/reader/books/9788533500426>>.

DEITEL, Paul. **Java: como programar.** 4. ed. Porto Alegre: Bookman, 2003.

DEITEL, Paul. **Java: como programar.** 10. ed. São Paulo: Pearson Education do Brasil, 2017.

Programação Orientada a Objetos – (POO0001)

JAVA

Alexandre Mendonça Fava
alexandre.fava@udesc.br

Universidade do Estado de Santa Catarina – UDESC
Programa de Pós-graduação em Computação Aplicada – PPGCA