

**INSTITUTO
FEDERAL**
Santa Catarina

Introdução ao Java

Pensamento Computacional e Algoritmos

Análise e Desenvolvimento de Sistemas

Prof. Adriano Lima

adriano.lima@ifsc.edu.br



cores

#111027

#277756

#16ABCD

#FFF4EC

#C74E23

fontes

Fira Sans Extra Condensed

Ubuntu

Roboto Mono





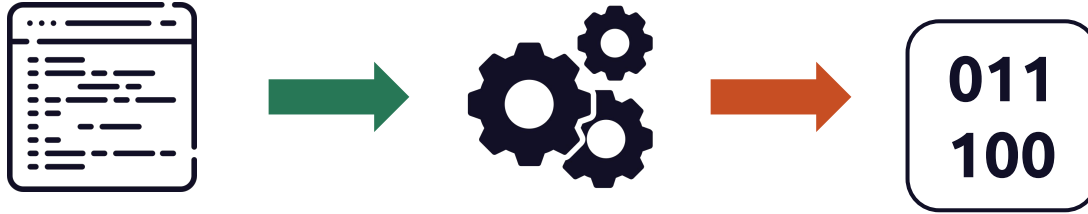
Java

- linguagem de programação de uso geral
- lançada em 1995
- última versão: Java SE 23 (set, 2024)
- 3ª linguagem mais utilizada (2022)¹
- linguagem compilada/interpretada
- escreva uma vez, rode em qualquer lugar
- base do Android

¹github.com

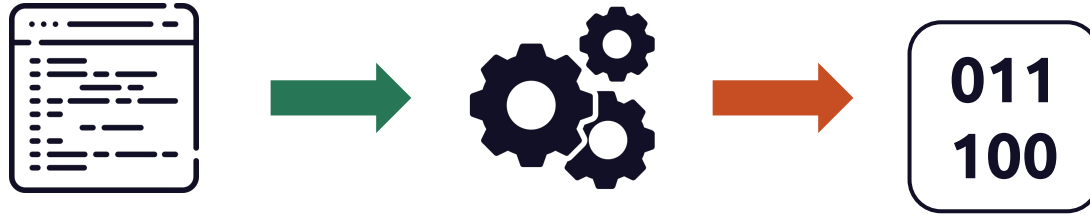
Compilador Java

compilador tradicional



Compilador Java

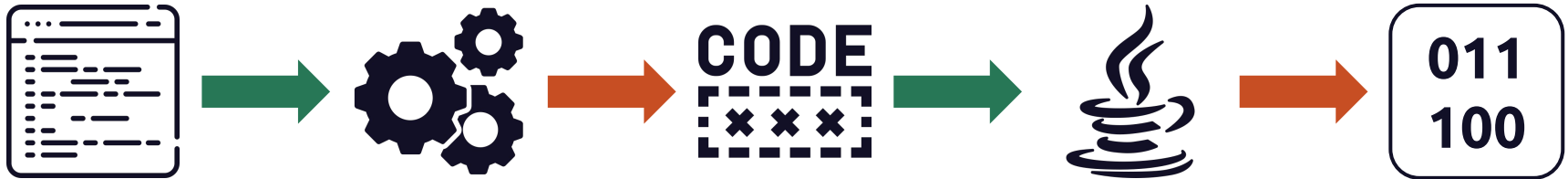
compilador tradicional



compilador Java

bytecode

jvm



Variáveis

- um símbolo ou nome significativo para um dado na memória ("rótulo")
- uma variável tem um valor associado
- o valor pode mudar durante a execução do programa
- variáveis pode ser de diferentes **tipos**

exemplo

nome da variável: "idade"
valor variável: 25



Tipos de Variáveis em Java

tipo	bits	exemplo
int	32	int idade = 25;
■ número inteiro		

Tipos de Variáveis em Java

tipo	bits	exemplo
int	32	int idade = 25;
char	16	char sexo = 'F';
■ caractere		

Tipos de Variáveis em Java

tipo	bits	exemplo
int	32	int idade = 25;
char	16	char sexo = 'F';
boolean	8	boolean achou = false;
■ valor booleano		

Tipos de Variáveis em Java

tipo	bits	exemplo
int	32	<code>int idade = 25;</code>
char	16	<code>char sexo = 'F';</code>
boolean	8	<code>boolean achou = false;</code>
long	64	<code>long bilhao = 1000000000;</code>
■ número inteiro longo		



Tipos de Variáveis em Java

tipo	bits	exemplo
int	32	int idade = 25;
char	16	char sexo = 'F';
boolean	8	boolean achou = false;
long	64	long bilhao = 1000000000;
float	32	float pi = 3.14f;

- número de ponto flutuante (número real)



Tipos de Variáveis em Java

tipo	bits	exemplo
int	32	<code>int idade = 25;</code>
char	16	<code>char sexo = 'F';</code>
boolean	8	<code>boolean achou = false;</code>
long	64	<code>long bilhao = 1000000000;</code>
float	32	<code>float pi = 3.14f;</code>
double	64	<code>double pi = 3.14159265358;</code>

- número de ponto flutuante longo

Tipos de Variáveis em Java

tipo	bits	exemplo
int	32	<code>int idade = 25;</code>
char	16	<code>char sexo = 'F';</code>
boolean	8	<code>boolean achou = false;</code>
long	64	<code>long bilhao = 1000000000;</code>
float	32	<code>float pi = 3.14f;</code>
double	64	<code>double pi = 3.14159265358;</code>
String	...	<code>String nome = "Maria";</code>

- sequência de caracteres

Variáveis

declarando uma variável

Variáveis

declarando uma variável

- **qualquer palavra pode ser usada, exceto as palavras reservadas (utilizadas pela linguagem)**



Variáveis

declarando uma variável

- qualquer palavra pode ser usada, exceto as palavras reservadas (utilizadas pela linguagem)
- **uma vez declarada, seu nome não muda**

Variáveis

declarando uma variável

- qualquer palavra pode ser usada, exceto as palavras reservadas (utilizadas pela linguagem)
- uma vez declarada, seu nome não muda
- **devem começar com uma letra, sublinhado (_) ou cifrão (\$)**



Variáveis

declarando uma variável

- qualquer palavra pode ser usada, exceto as palavras reservadas (utilizadas pela linguagem)
- uma vez declarada, seu nome não muda
- devem começar com uma letra, sublinhado (_) ou cifrão (\$)
- **por convenção, utilizam-se letras minúsculas**



Variáveis

declarando uma variável

- qualquer palavra pode ser usada, exceto as palavras reservadas (utilizadas pela linguagem)
- uma vez declarada, seu nome não muda
- devem começar com uma letra, sublinhado (_) ou cifrão (\$)
- por convenção, utilizam-se letras minúsculas
- **letras maiúsculas são usada para iniciar outra palavra (ex. nomeCliente)**



Variáveis

declarando uma variável

- qualquer palavra pode ser usada, exceto as palavras reservadas (utilizadas pela linguagem)
- uma vez declarada, seu nome não muda
- devem começar com uma letra, sublinhado (_) ou cifrão (\$)
- por convenção, utilizam-se letras minúsculas
- letras maiúsculas são usada para iniciar outra palavra (ex. nomeCliente)



Palavras Reservadas em Java

abstract assert boolean break byte case catch char
class const continue default do double else enum
extends final finally float for goto if implements
import instanceof int interface long native new
package private protected public return short
static strictfp super switch synchronized this
throw throws transient try void volatile while var
true null false



Palavras Reservadas em Java

abstract assert **boolean** break byte case catch **char**
class const continue default do **double** else enum
extends final finally **float** for goto if implements
import instanceof **int** interface **long** native new
package private protected public return short
static strictfp super switch synchronized this
throw throws transient try void volatile while var
true null false



Constantes

declarando uma constante

- **variável que não pode ter seu valor alterado**

Constantes

declarando uma constante

- variável que não pode ter seu valor alterado
- **é declarada de forma semelhante a uma variável, mas com o termo `final` antes do tipo**



Constantes

declarando uma constante

- variável que não pode ter seu valor alterado
- é declarada de forma semelhante a uma variável, mas com o termo `final` antes do tipo
- **devem começar com uma letra, sublinhado (`_`) ou cifrão (`$`)**



Constantes

declarando uma constante

- variável que não pode ter seu valor alterado
- é declarada de forma semelhante a uma variável, mas com o termo `final` antes do tipo
- devem começar com uma letra, sublinhado (`_`) ou cifrão (`$`)
- **por convenção, utilizam-se letras maiúsculas**



Constantes

declarando uma constante

- variável que não pode ter seu valor alterado
- é declarada de forma semelhante a uma variável, mas com o termo `final` antes do tipo
- devem começar com uma letra, sublinhado (`_`) ou cifrão (`$`)
- por convenção, utilizam-se letras maiúsculas

exemplo `final float PI = 3.14f;`



Operadores Aritméticos

operador	operação	exemplo
+	adição	$x + y$

Operadores Aritméticos

operador	operação	exemplo
+	adição	$x + y$
-	subtração	$x - y$

Operadores Aritméticos

operador	operação	exemplo
+	adição	$x + y$
-	subtração	$x - y$
*	multiplicação	$x * y$

Operadores Aritméticos

operador	operação	exemplo
+	adição	$x + y$
-	subtração	$x - y$
*	multiplicação	$x * y$
/	divisão	x / y

Operadores Aritméticos

operador	operação	exemplo
+	adição	$x + y$
-	subtração	$x - y$
*	multiplicação	$x * y$
/	divisão	x / y
%	resto da divisão	$x \% y$

Operadores Aritméticos

operador	operação	exemplo
+	adição	$x + y$
-	subtração	$x - y$
*	multiplicação	$x * y$
/	divisão	x / y
%	resto da divisão	$x \% y$
++	incremento unitário	$x++$ ou $++x$

Operadores Aritméticos

operador	operação	exemplo
+	adição	$x + y$
-	subtração	$x - y$
*	multiplicação	$x * y$
/	divisão	x / y
%	resto da divisão	$x \% y$
++	incremento unitário	$x++$ ou $++x$
--	decremento unitário	$x--$ ou $--x$

Operadores Aritméticos

operador	operação	exemplo
+	adição	$x + y$
-	subtração	$x - y$
*	multiplicação	$x * y$
/	divisão	x / y
%	resto da divisão	$x \% y$
++	incremento unitário	$x++$ ou $++x$
--	decremento unitário	$x--$ ou $--x$

Operadores Relacionais

operador	operação	exemplo
==	igualdade	x == y



Operadores Relacionais

operador	operação	exemplo
==	igualdade	x == y
!=	diferença	x != y

Operadores Relacionais

operador	operação	exemplo
==	igualdade	$x == y$
!=	diferença	$x != y$
>	maior que	$x > y$

Operadores Relacionais

operador	operação	exemplo
==	igualdade	$x == y$
!=	diferença	$x != y$
>	maior que	$x > y$
>=	maior que ou igual a	$x >= y$



Operadores Relacionais

operador	operação	exemplo
==	igualdade	$x == y$
!=	diferença	$x != y$
>	maior que	$x > y$
>=	maior que ou igual a	$x >= y$
<	menor que	$x < y$



Operadores Relacionais

operador	operação	exemplo
==	igualdade	$x == y$
!=	diferença	$x != y$
>	maior que	$x > y$
>=	maior que ou igual a	$x >= y$
<	menor que	$x < y$
<=	menor que ou igual a	$x <= y$



Operadores Relacionais

operador	operação	exemplo
==	igualdade	$x == y$
!=	diferença	$x != y$
>	maior que	$x > y$
>=	maior que ou igual a	$x >= y$
<	menor que	$x < y$
<=	menor que ou igual a	$x <= y$

Operadores Lógicos

operador	operação	exemplo
&&	E lógico (AND)	x && y

Operadores Lógicos

operador	operação	exemplo
&&	E lógico (AND)	x && y
	OU lógico (OR)	x y

Operadores Lógicos

operador	operação	exemplo
&&	E lógico (AND)	x && y
	OU lógico (OR)	x y
!	negação (NOT)	!x

Operadores Lógicos

operador	operação	exemplo
&&	E lógico (AND)	x && y
	OU lógico (OR)	x y
!	negação (NOT)	!x

x	!x
true	false
true	false
false	true
false	true

Operadores Lógicos

operador	operação	exemplo
&&	E lógico (AND)	x && y
	OU lógico (OR)	x y
!	negação (NOT)	!x

x	y	!x	x y
true	true	false	true
true	false	false	true
false	true	true	true
false	false	true	false

Operadores Lógicos

operador	operação	exemplo
&&	E lógico (AND)	x && y
	OU lógico (OR)	x y
!	negação (NOT)	!x

x	y	!x	x y	x && y
true	true	false	true	true
true	false	false	true	false
false	true	true	true	false
false	false	true	false	false

Operadores Lógicos

operador	operação	exemplo
&&	E lógico (AND)	x && y
	OU lógico (OR)	x y
!	negação (NOT)	!x

x	y	!x	x y	x && y
true	true	false	true	true
true	false	false	true	false
false	true	true	true	false
false	false	true	false	false

Exercício 1

1. Declare três variáveis numéricas (idade, ano atual e ano de nascimento).
2. Atribua a sua idade à primeira variável e o ano atual à segunda variável.
3. Calcule o seu ano de nascimento e atribua à terceira variável.

Exercício 2

1. Declare três variáveis numéricas (tempo para estudo, número de disciplinas e tempo por disciplina).
2. Atribua o tempo que você tem disponível para estudo à primeira variável e o número de disciplinas que está cursando à segunda variável.
3. Calcule quanto tempo você tem para estudar cada disciplina.



Exercício 3

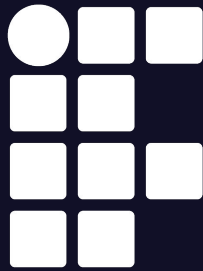
1. Declare uma variável numérica (tempo ideal de estudo) e atribua o valor 4 a ela.
2. Teste se o tempo que você tem por disciplina é maior ou menor do que o tempo ideal para estudo.



Exercício 4

1. Determine o valor para cada expressão

	Expressão	A	B	C	D	Valor
1	$(A > B) \text{ E } (C > D)$	10	8	20	15	
2	$(A > B) \text{ E } (C > D)$	10	12	15	15	
3	$(A \leq B) \text{ OU } (C > 10)$	5	5	3	5	
4	$(A \leq B) \text{ OU } ((C == D) \text{ E } (D < A))$	3	10	4	4	
5	$(A == B) \text{ OU } (C == D)$	3	1	4	4	
6	$((A == B) \text{ E } (B < 30)) \text{ OU } (C >= D)$	5	5	10	5	



**INSTITUTO
FEDERAL**
Santa Catarina

Introdução ao Java

Pensamento Computacional e Algoritmos

Análise e Desenvolvimento de Sistemas

Prof. Adriano Lima

adriano.lima@ifsc.edu.br

