

# **FP61A**

# **FUNDAMENTOS DE**

# **PROGRAMAÇÃO**

## **Aula 03 - Fluxogramas**

**Profs. Rafael Mantovani**



Apucarana - PR, Brasil

Universidade Tecnológica Federal do Paraná (UTFPR)

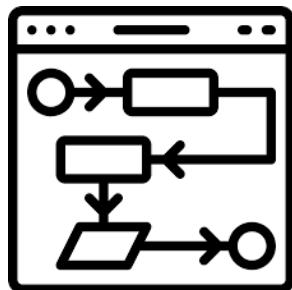
# Roteiro

- 1 Introdução**
- 2 Fluxograma**
- 3 Variáveis**
- 4 Operadores e Estruturas de Controle**
- 5 Exercícios**
- 6 Referências**

# Roteiro

- 1 Introdução**
- 2 Fluxograma**
- 3 Variáveis**
- 4 Operadores e Estruturas de Controle**
- 5 Exercícios**
- 6 Referências**

# Algoritmo



... são conjuntos de passos **finitos** e **organizados** que, quando executados, **resolvem um determinado problema**.

# Fluxo



# Representações

**Os três tipos mais utilizados de representação de algoritmos são:**

# Representações

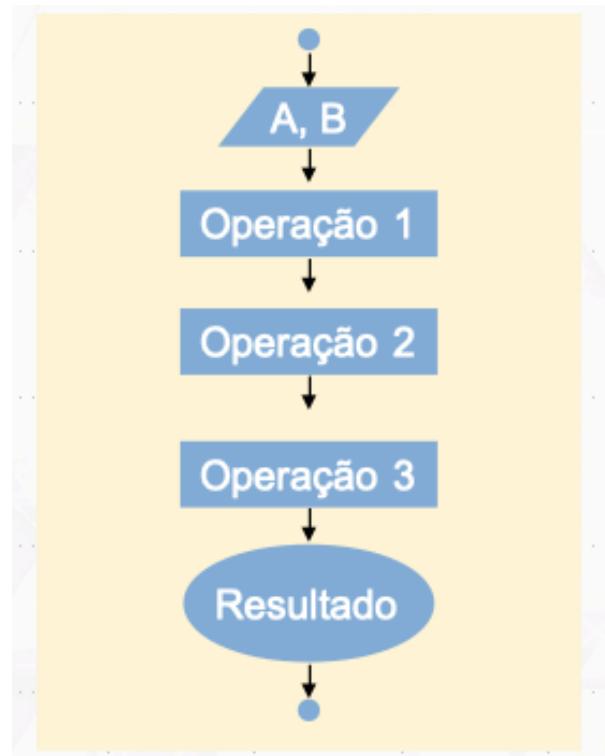
**Os três tipos mais utilizados de representação de algoritmos são:**

1. Receber os dados
2. Executar operação 1
3. Executar operação 2
4. Executar operação 3
5. Mostrar o resultado

**Narrativa**

# Representações

**Os três tipos mais utilizados de representação de algoritmos são:**



**Fluxograma**

# Representações

Os três tipos mais utilizados de representação de algoritmos são:

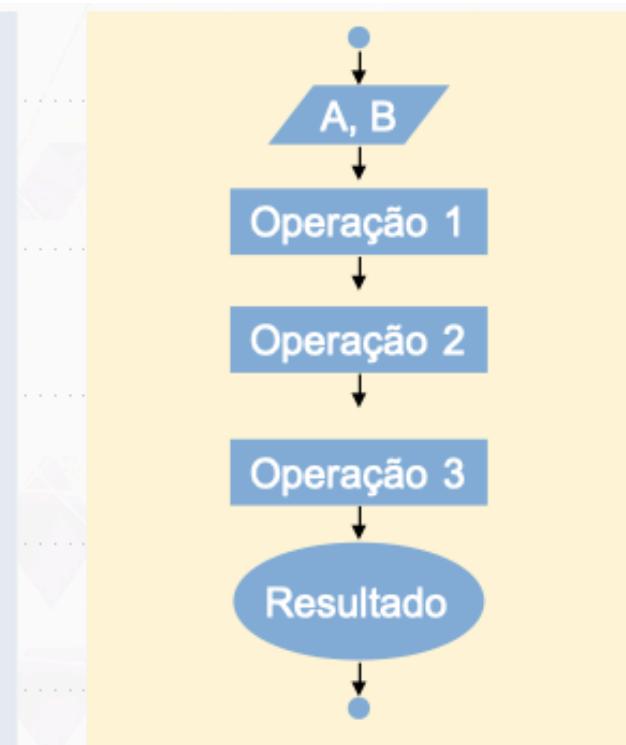
```
Algoritmo Operacoes
  Inicio
    Inteiro: A, B, resultado;
    Operacao 1;
    Operacao 2;
    Operacao 3;
    escreva(resultado);
  Fim
```

Pseucodcódigo  
(Portugol)

# Representações

Os três tipos mais utilizados de representação de algoritmos são:

1. Receber os dados
2. Executar operação 1
3. Executar operação 2
4. Executar operação 3
5. Mostrar o resultado



Narrativa

Fluxograma

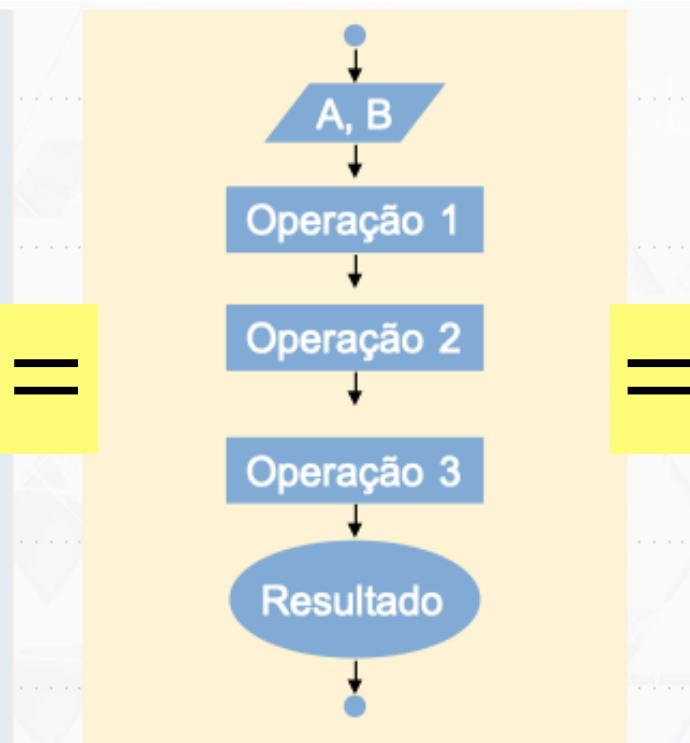
Pseucodcódigo  
(Portugol)

Algoritmo Operacoes  
Início  
 Inteiro: A, B, resultado;  
 Operacao 1;  
 Operacao 2;  
 Operacao 3;  
 escreva(resultado);  
Fim

# Representações

Os três tipos mais utilizados de representação de algoritmos são:

1. Receber os dados
2. Executar operação 1
3. Executar operação 2
4. Executar operação 3
5. Mostrar o resultado



Narrativa

Fluxograma

Pseucodcódigo  
(Portugol)

Algoritmo Operacoes  
Início  
Inteiro: A, B, resultado;  
Operacao 1;  
Operacao 2;  
Operacao 3;  
escreva(resultado);  
Fim

# Roteiro

- 1 Introdução**
- 2 Fluxograma**
- 3 Variáveis**
- 4 Operadores e Estruturas de Controle**
- 5 Exercícios**
- 6 Referências**

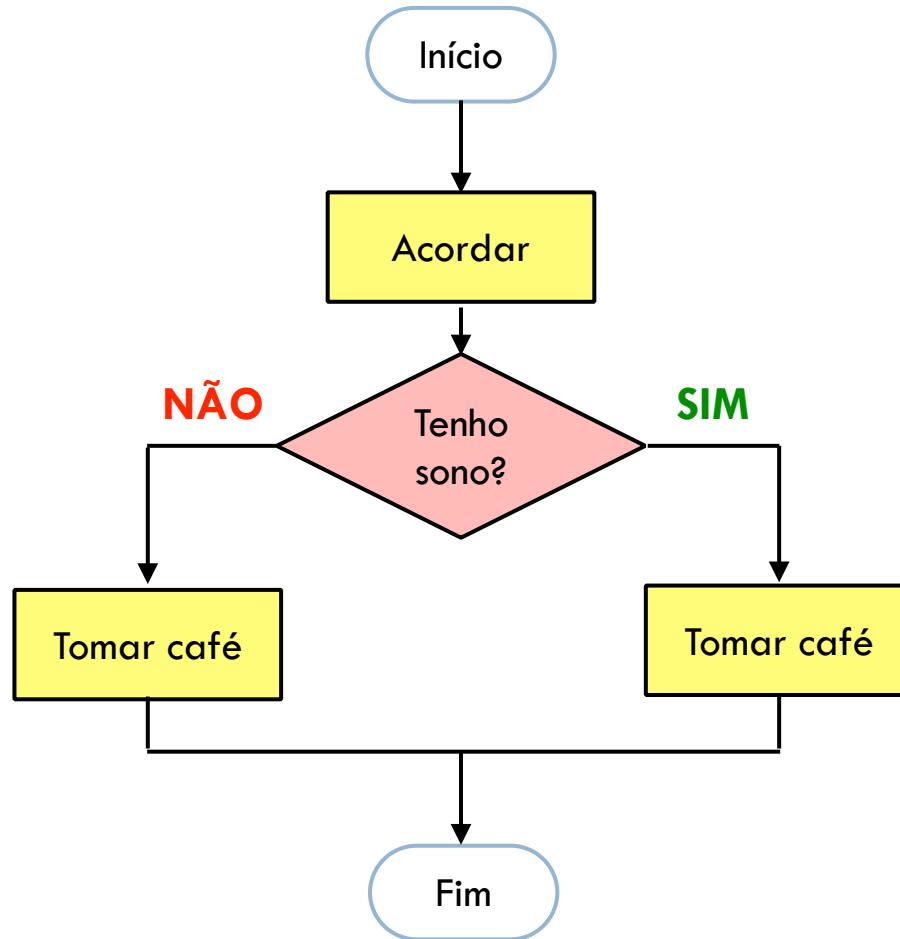
# Fluxograma



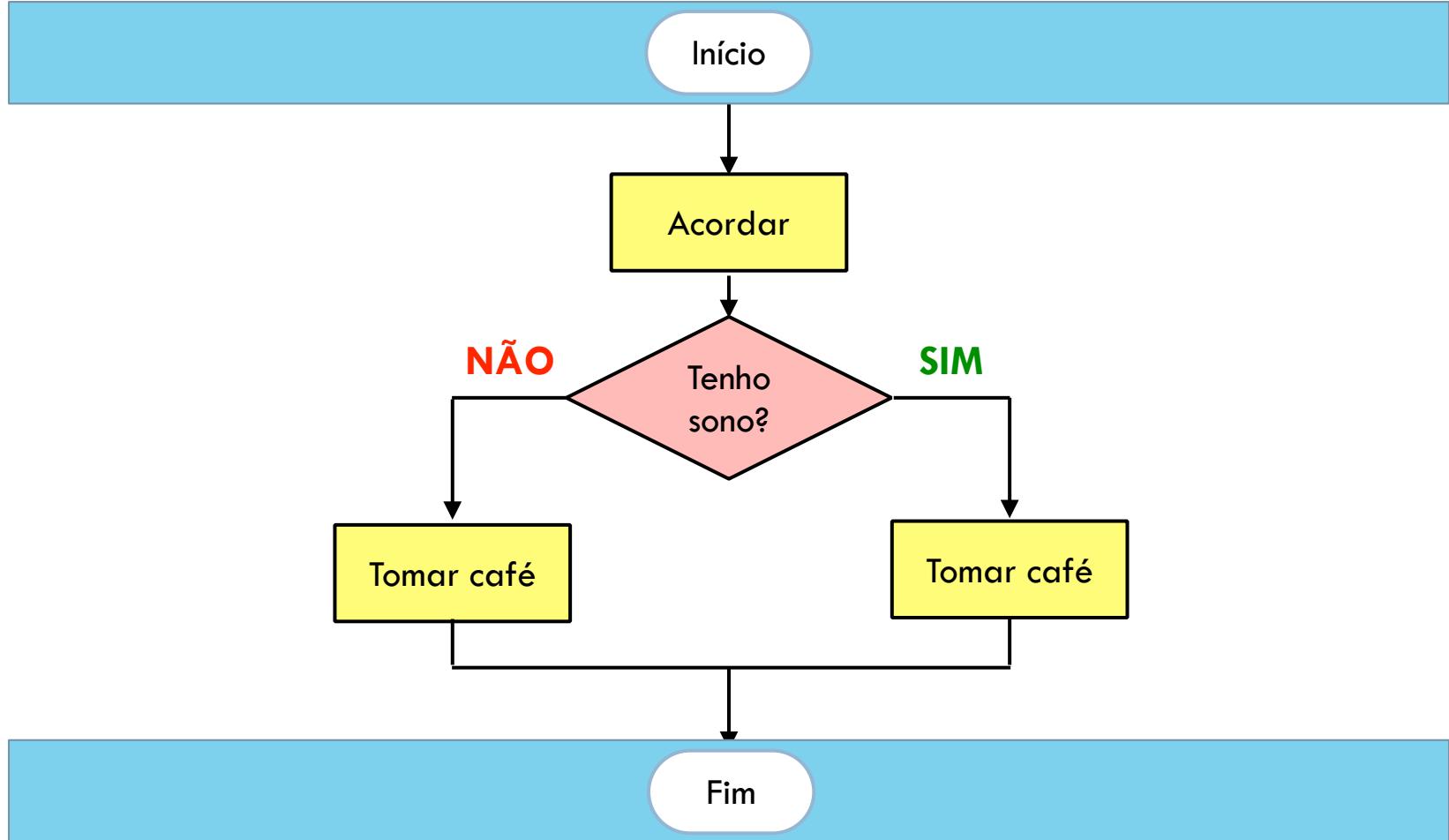
# Fluxograma



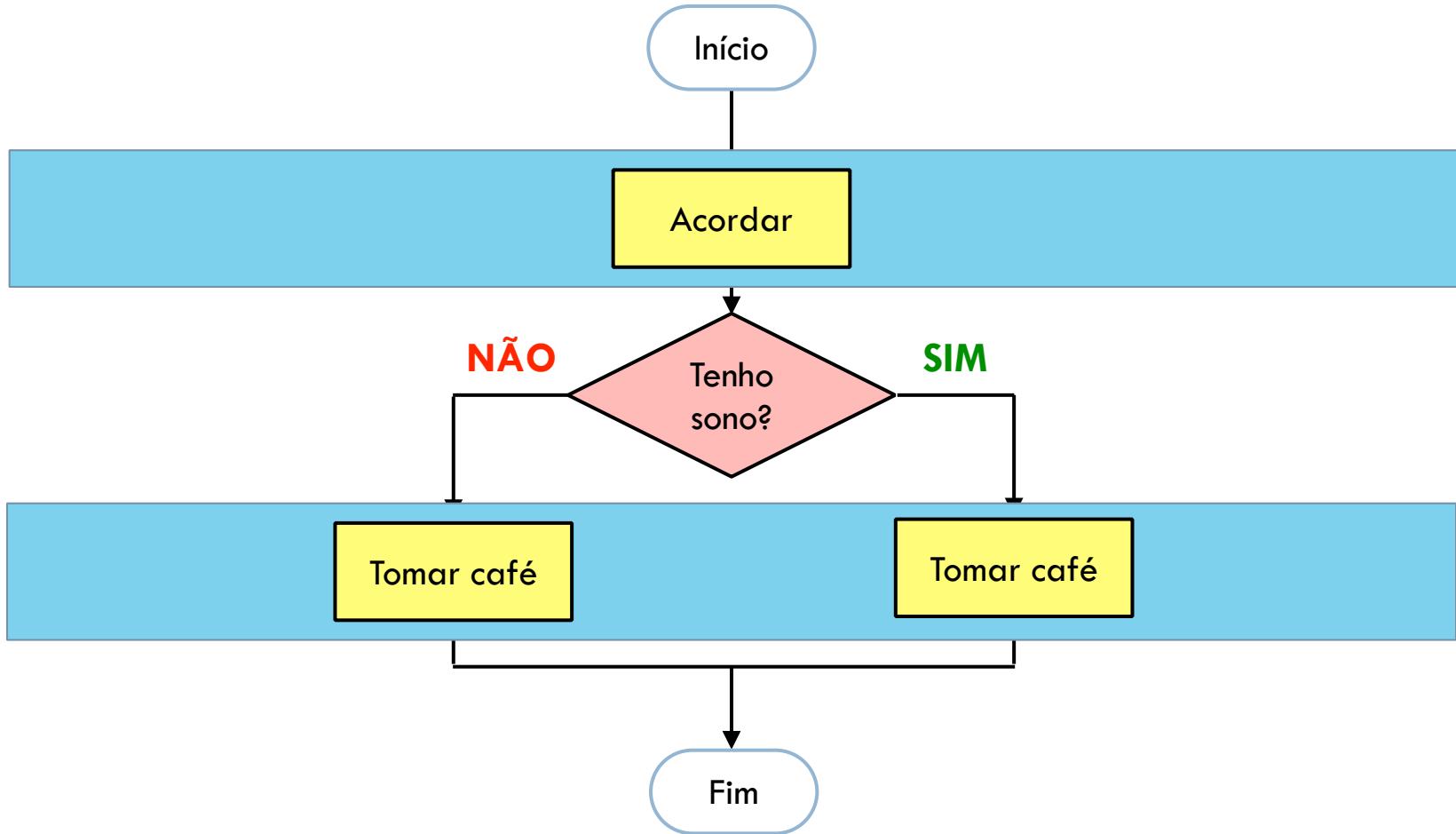
# Fluxograma



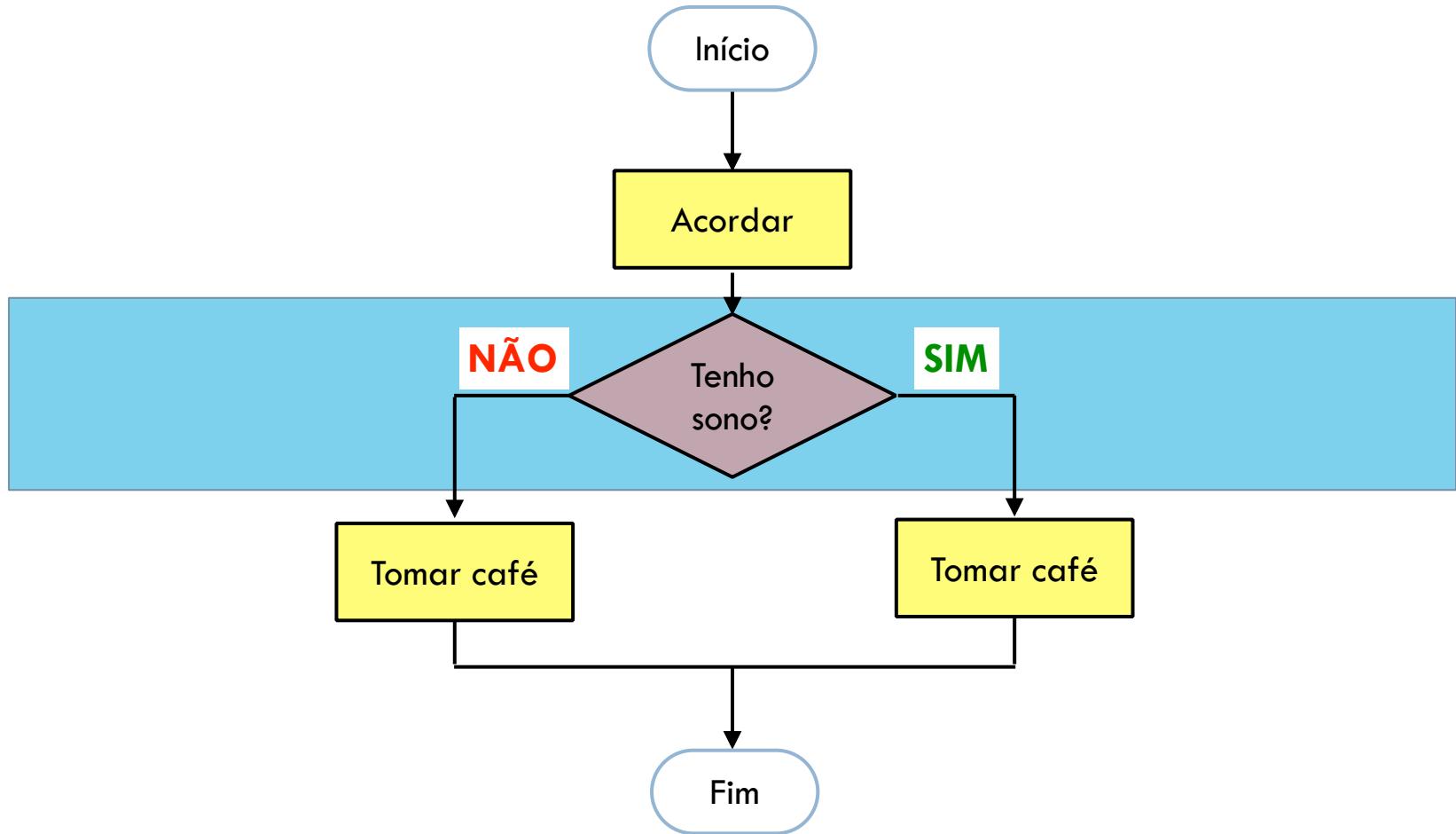
# Fluxograma



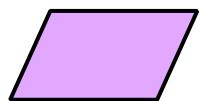
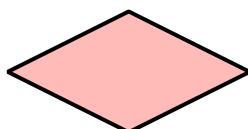
# Fluxograma



# Fluxograma



# Fluxograma



# Fluxograma



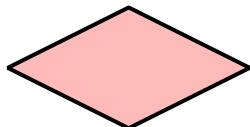
TERMINAL



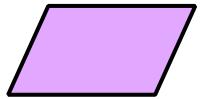
TECLADO



PROCESSAMENTO



DECISÃO



DISPLAY



SETA

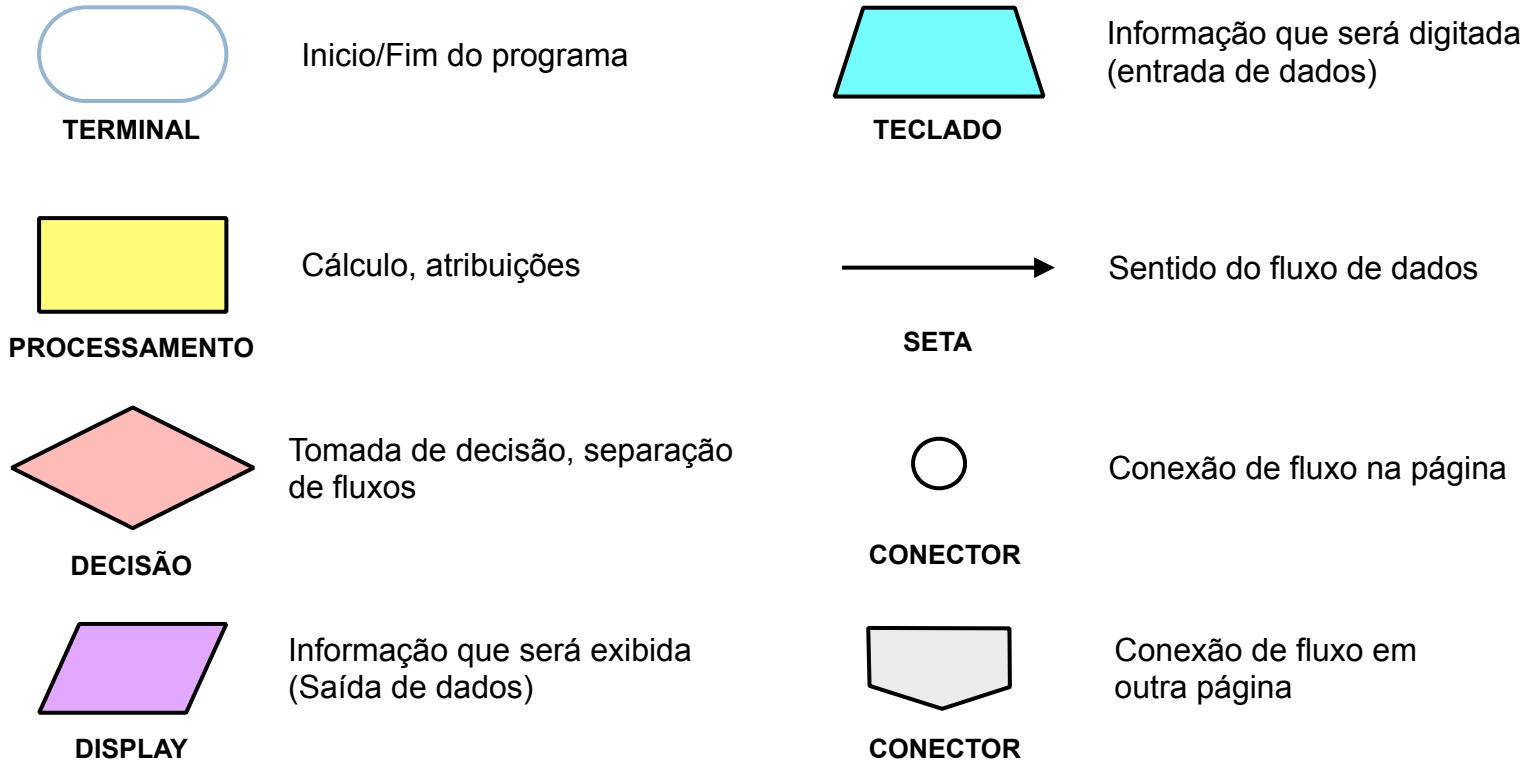


CONECTOR

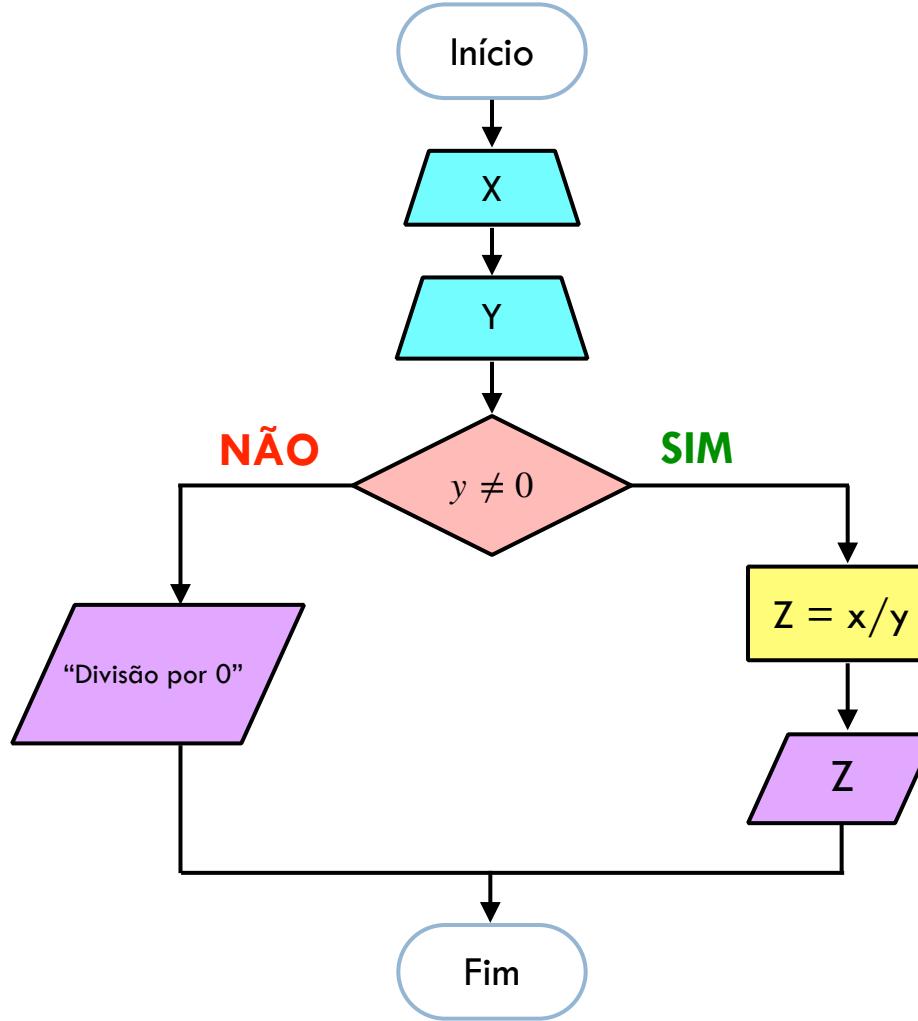


CONECTOR

# Fluxograma

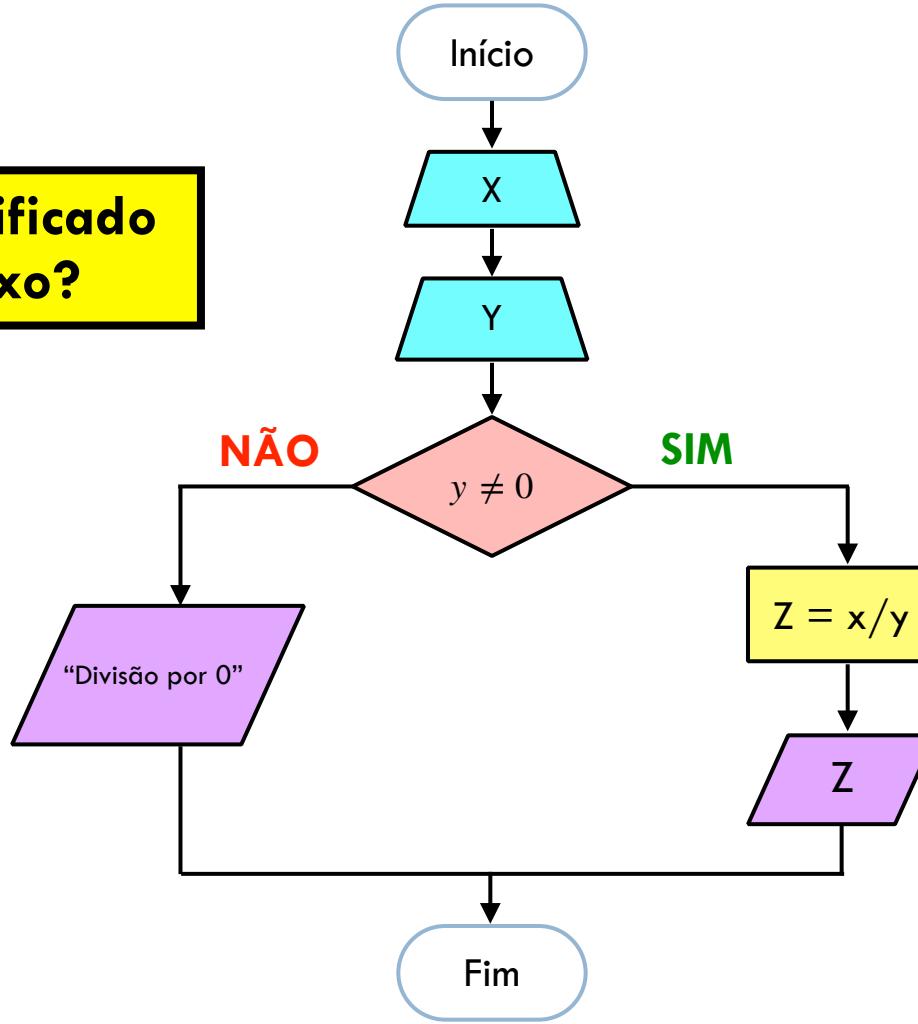


# Fluxograma



# Fluxograma

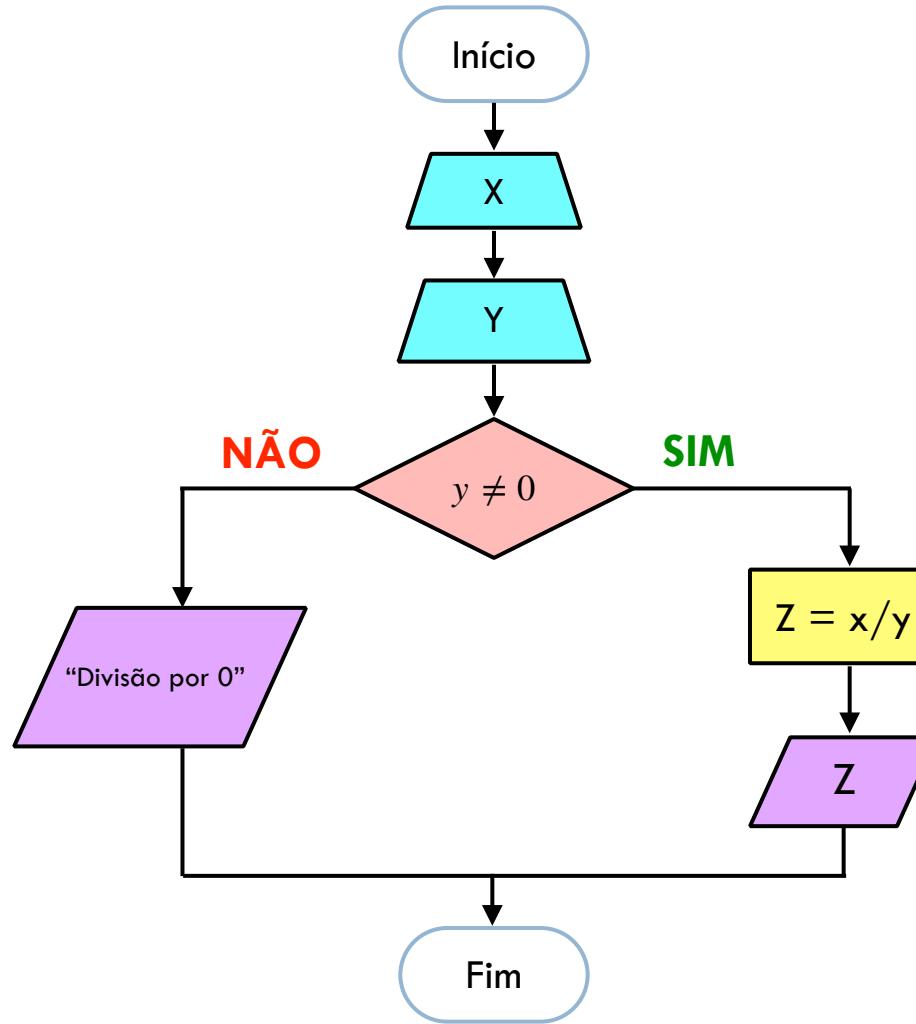
**Qual o significado  
deste fluxo?**



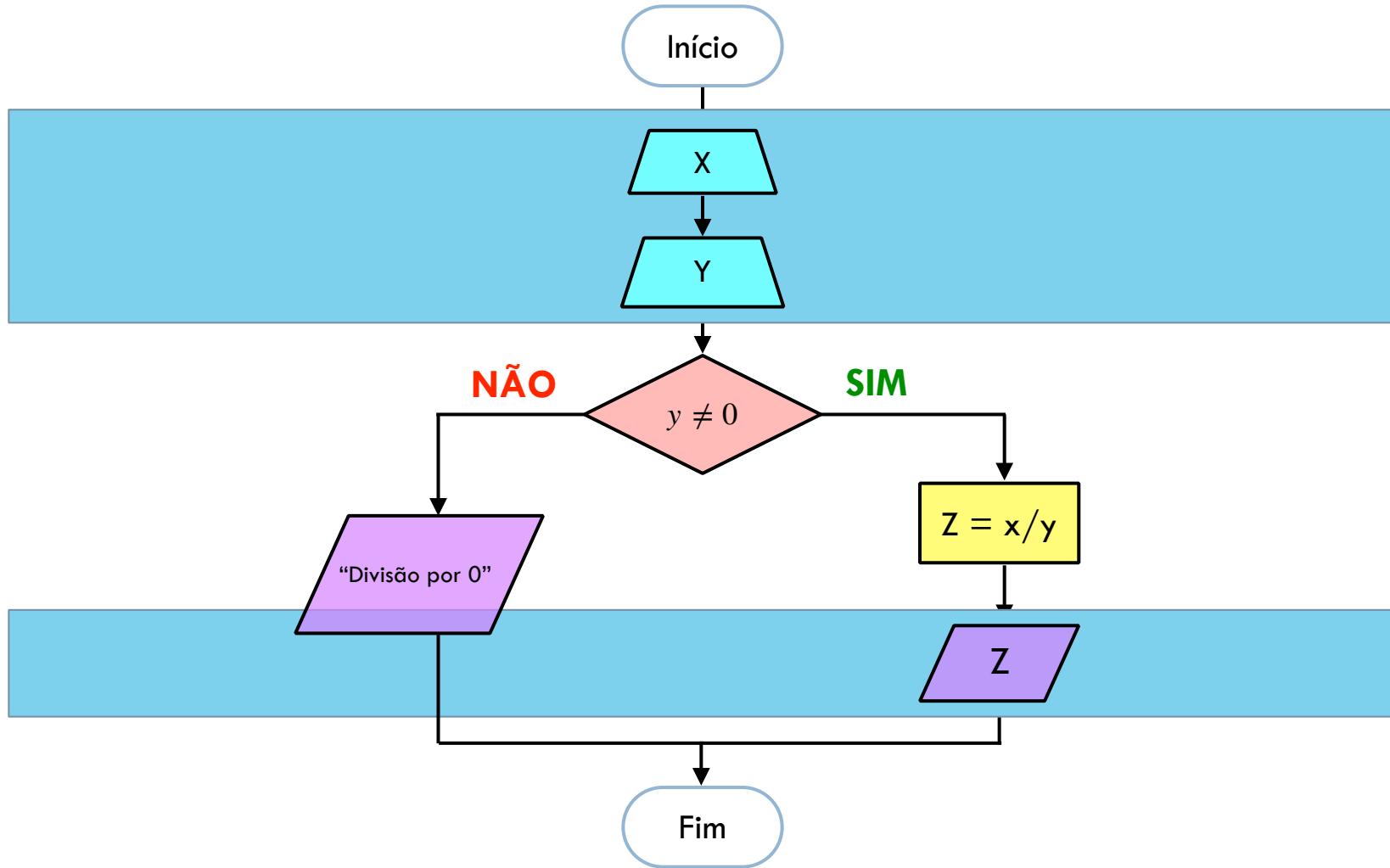
# Roteiro

- 1** Introdução
- 2** Fluxograma
- 3** Variáveis
- 4** Operadores e Estruturas de Controle
- 5** Exercícios
- 6** Referências

# Variáveis



# Variáveis



# Variáveis

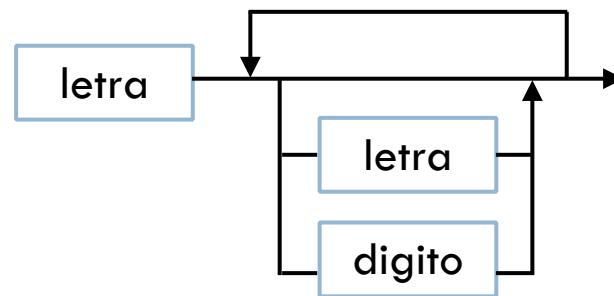
- Uma variável é uma localização na memória do computador que é utilizada para **armazenar temporariamente** os dados que são **utilizados pelo programa**
- As variáveis possuem algumas características, como:
  - nome (identificação da variável)
  - Tipo
  - Endereço
  - Tamanho
  - Valor (conteúdo da variável)

# Variáveis



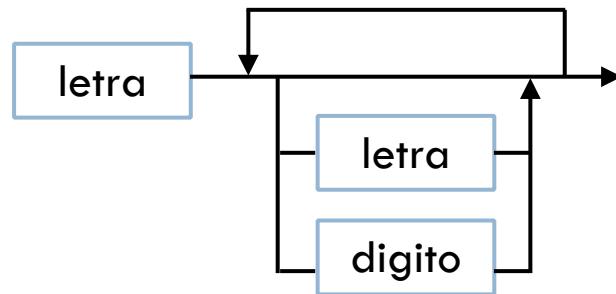
# Variáveis

## Tipo Identificador



# Variáveis

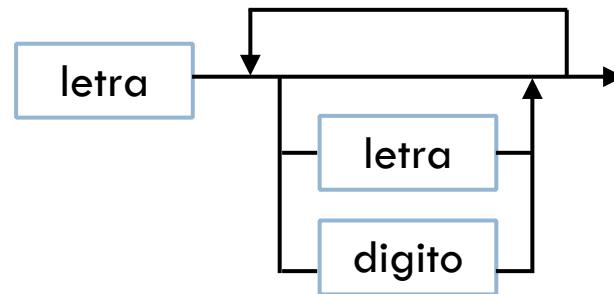
## Tipo Identificador



- Identificadores (nomes)
  - devem começar com uma **letra**
  - os próximos caracteres podem ser **letras** ou **números**
  - não pode usar nenhum símbolo, exceto **underline (\_)**
  - não pode conter **espaços em branco**
  - não pode conter **letras com acentos**

# Variáveis

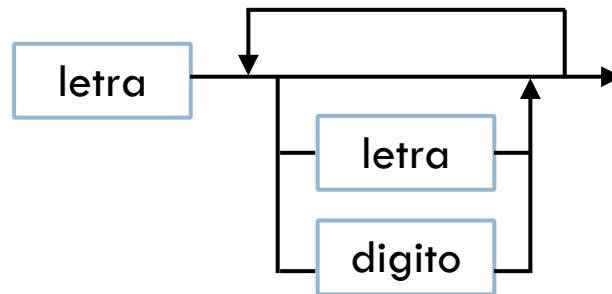
## Tipo Identificador



- Tipos:
  - Inteiro (**int**):
  - Real (**float** - ponto flutuante):
  - Caractere (**char** - character):
  - Lógico (**bool** - booleano):

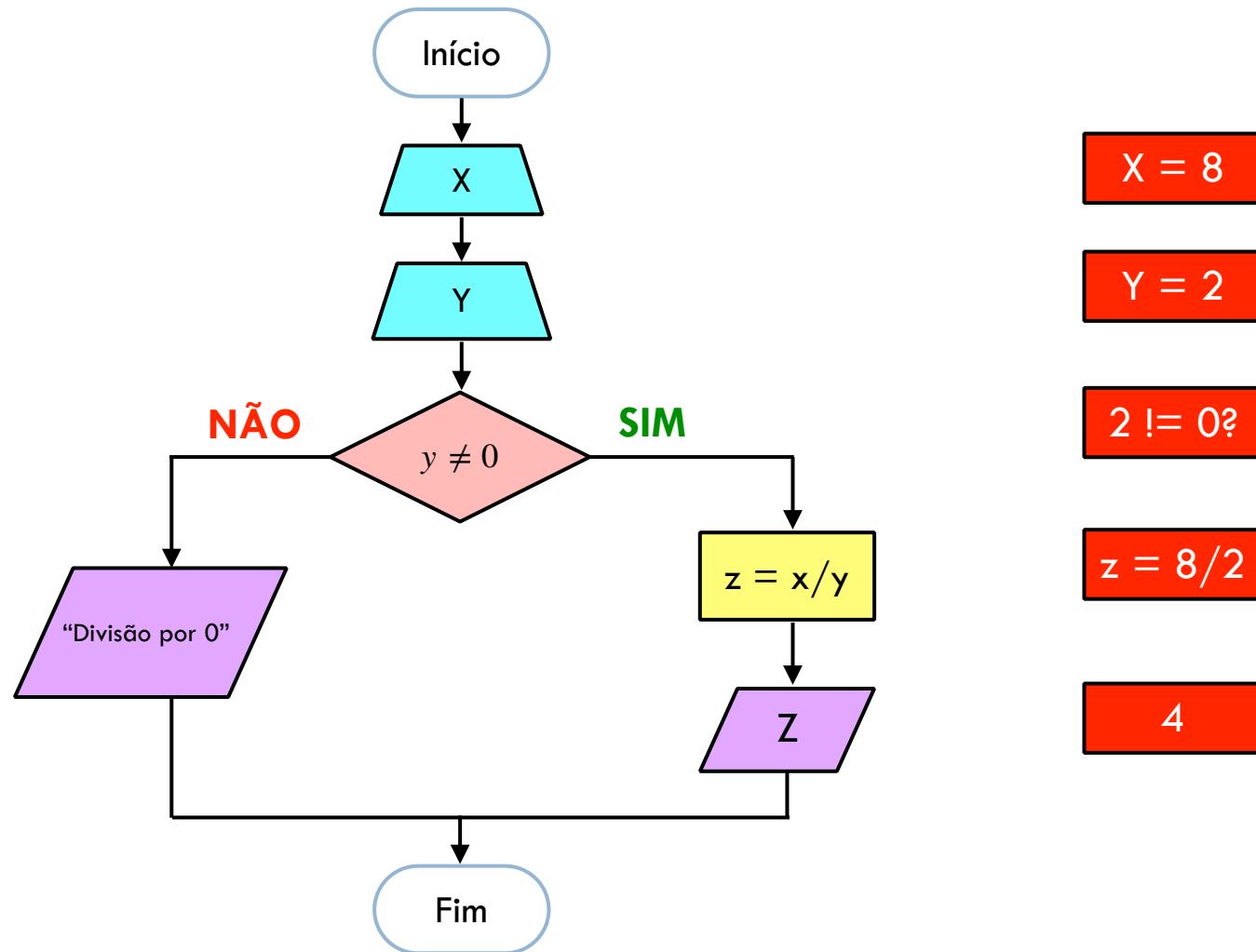
# Variáveis

## Tipo Identificador

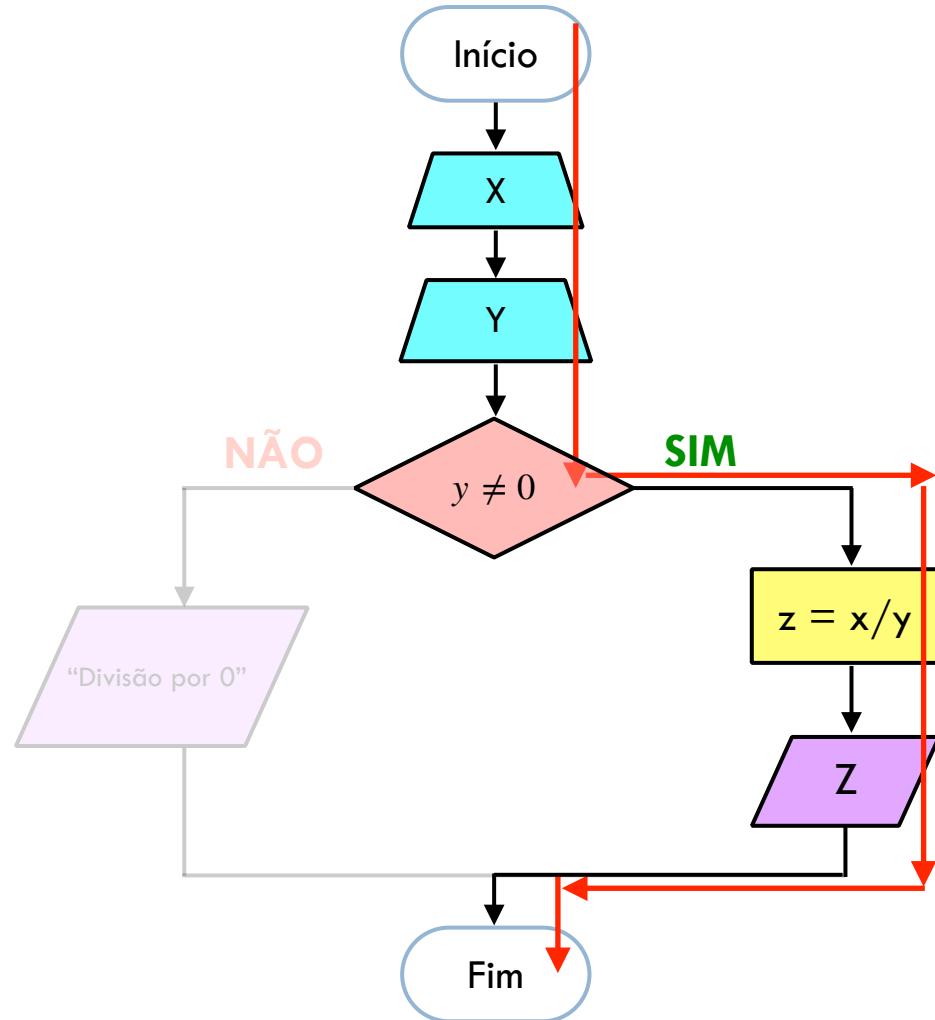


- Tipos:
  - Inteiro (**int**): 1 3 -5 198 0
  - Real (**float** - ponto flutuante): 0.5 5.0 9.8 -77.3 3.1415
  - Caractere (**char** - character): "programa" 'c' "algoritmo" "123"
  - Lógico (**bool** - booleano): verdadeiro/falso

# Variáveis



# Variáveis



X = 8

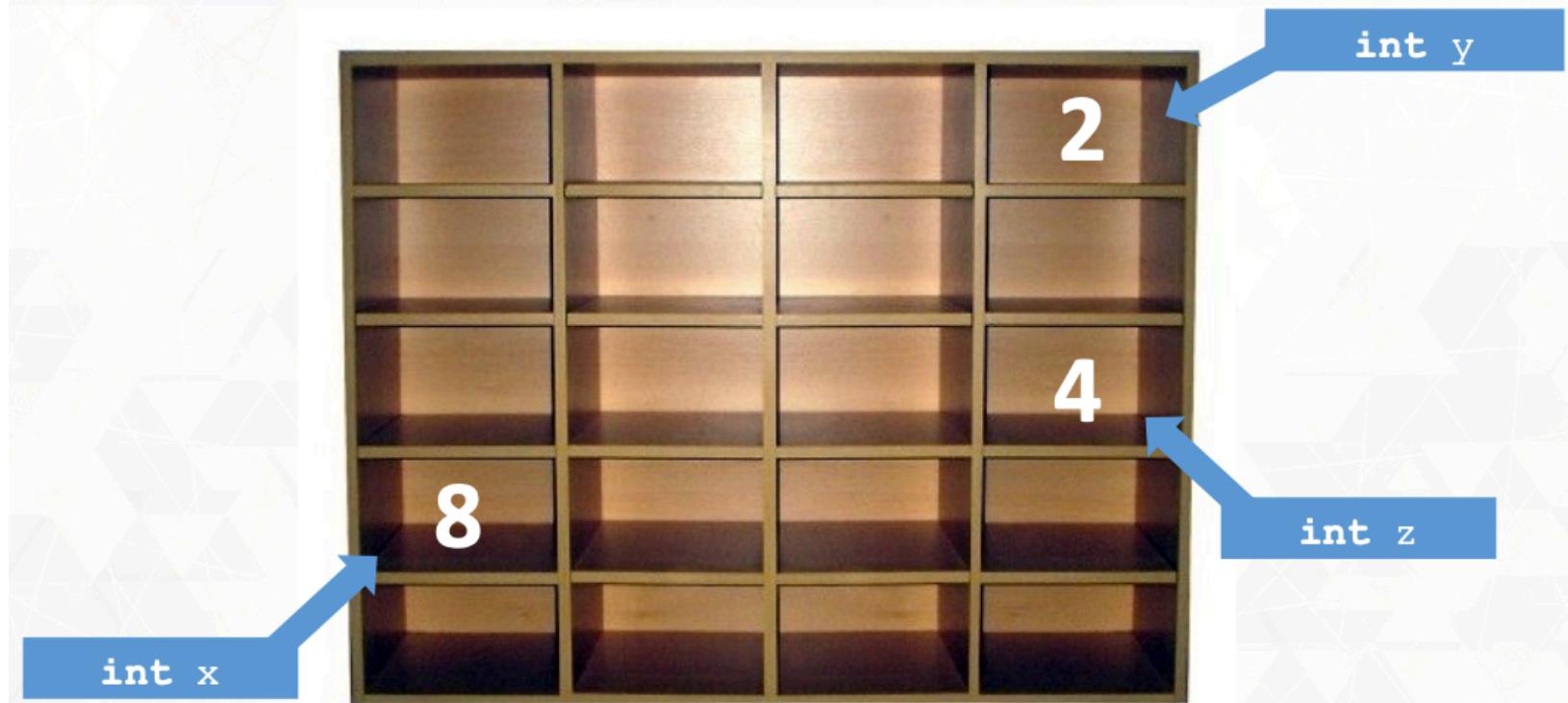
Y = 2

2 != 0?

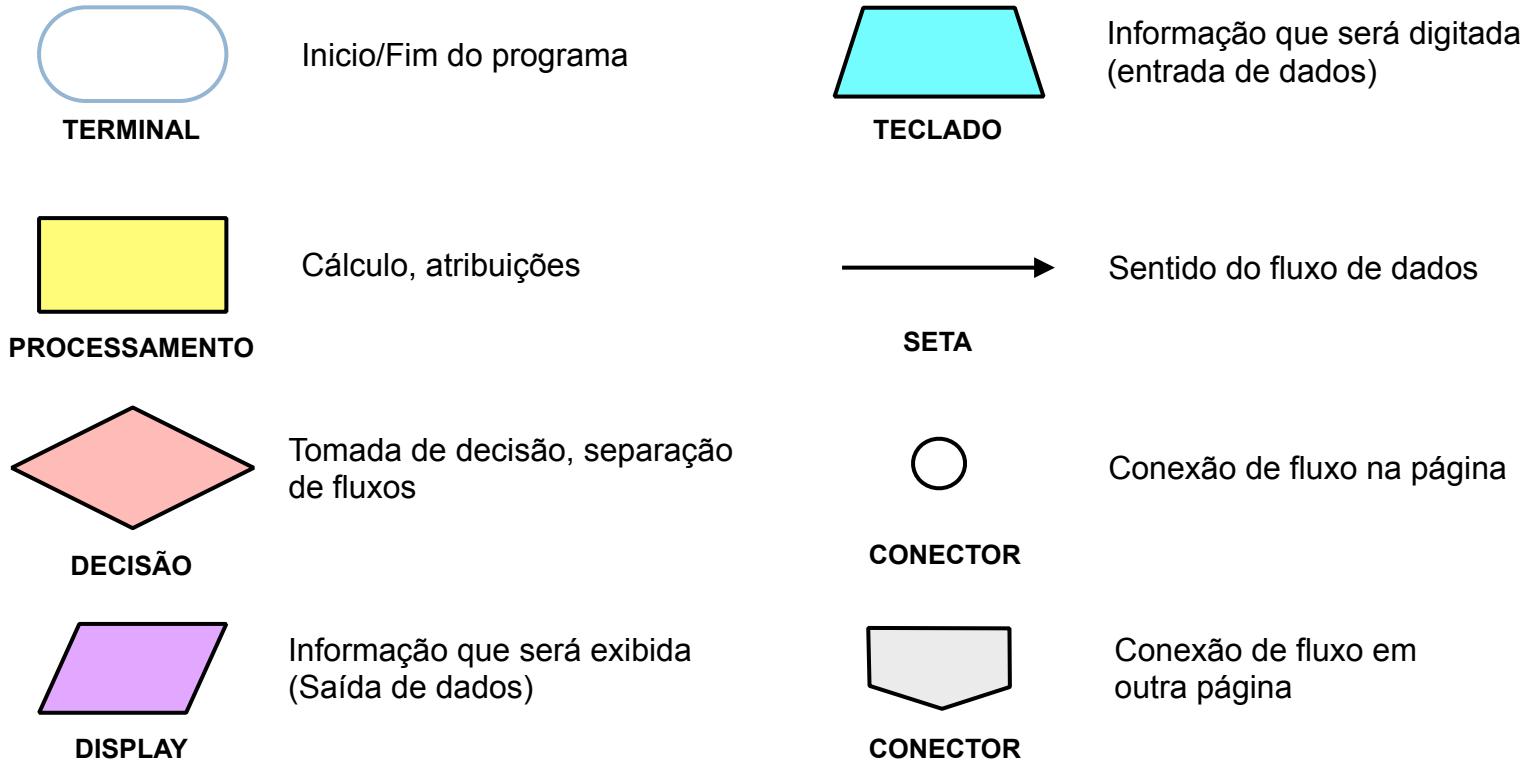
$z = 8/2$

4

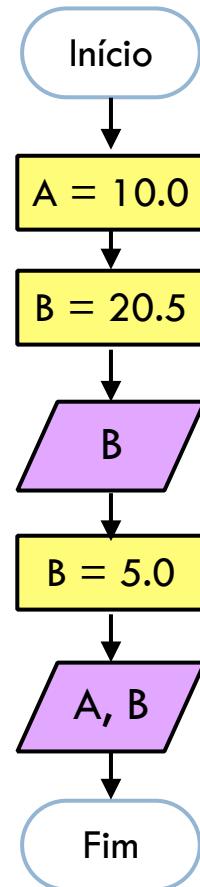
# Variáveis



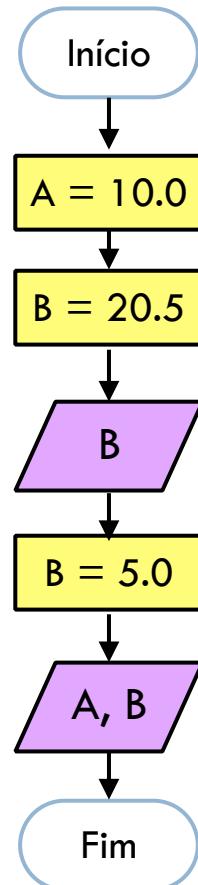
# Fluxograma



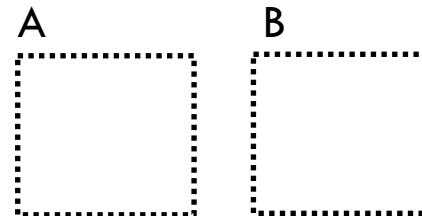
# Exemplo



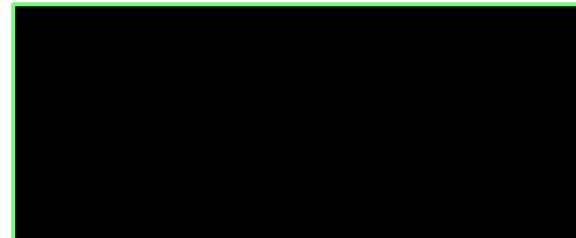
# Exemplo



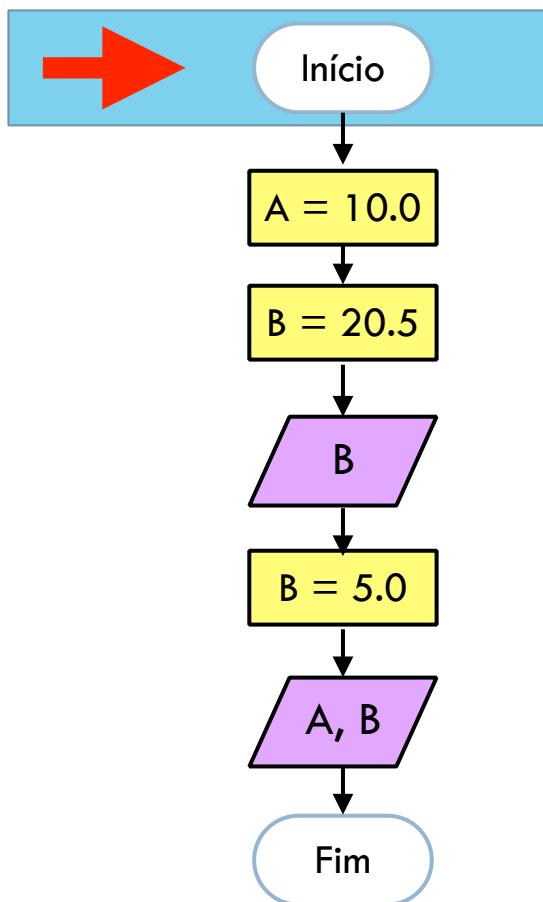
**Memória**



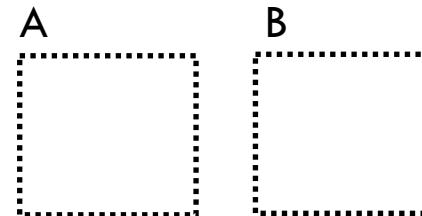
**O que o usuário ve?**



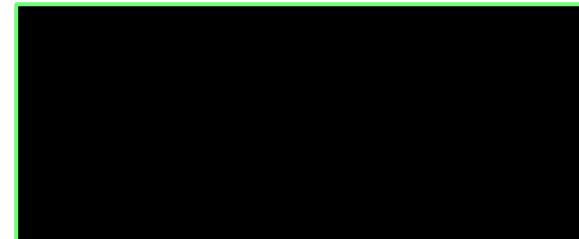
# Exemplo



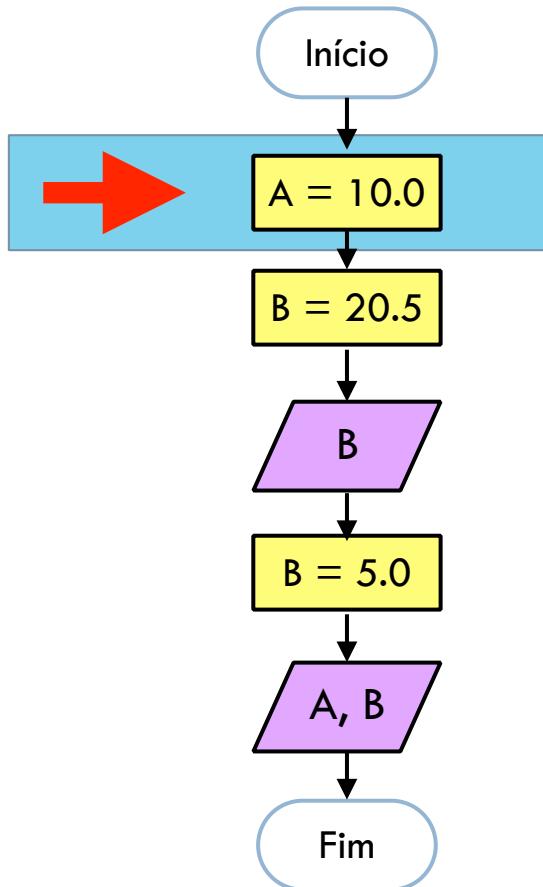
**Memória**



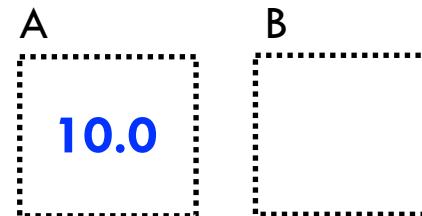
**O que o usuário ve?**



# Exemplo



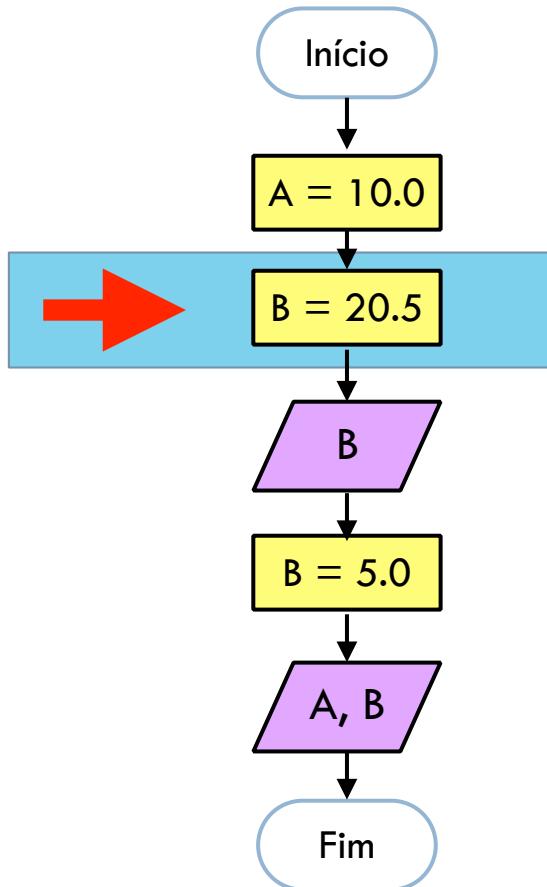
**Memória**



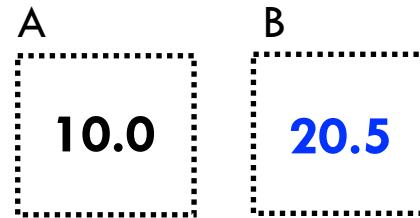
**O que o usuário ve?**



# Exemplo



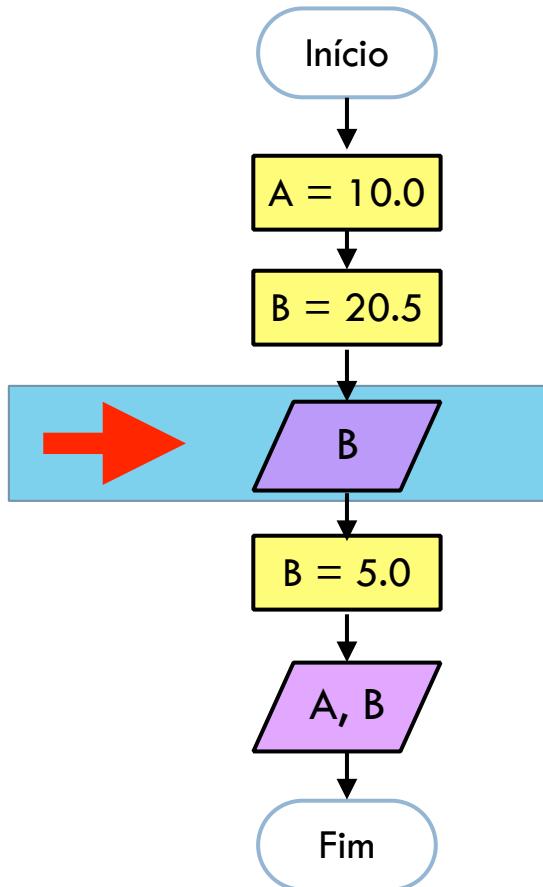
**Memória**



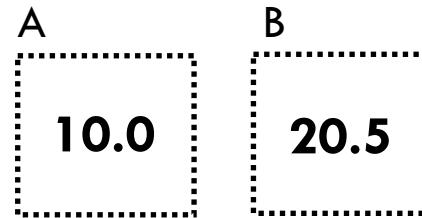
**O que o usuário ve?**



# Exemplo



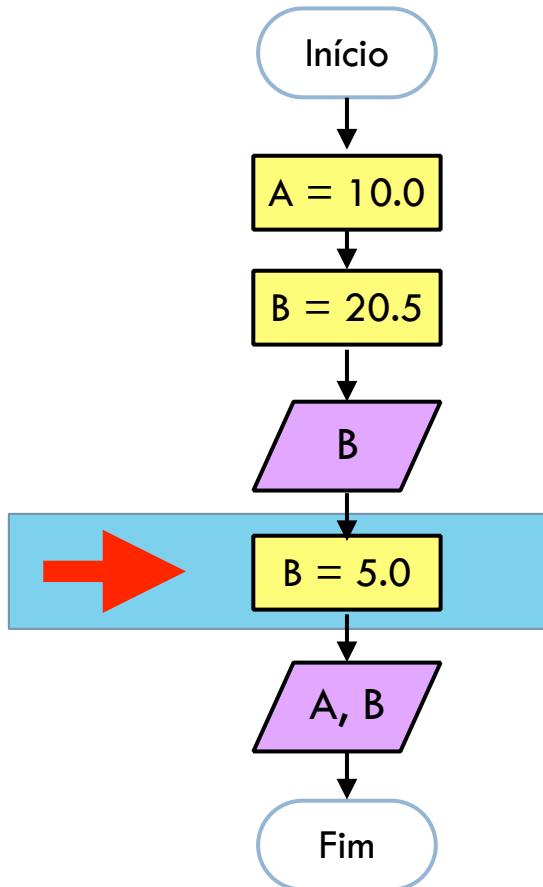
**Memória**



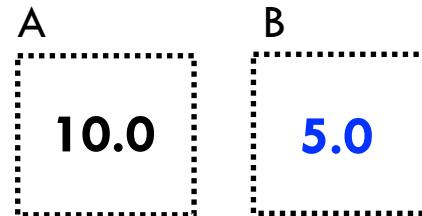
**O que o usuário ve?**

>> 20.5

# Exemplo



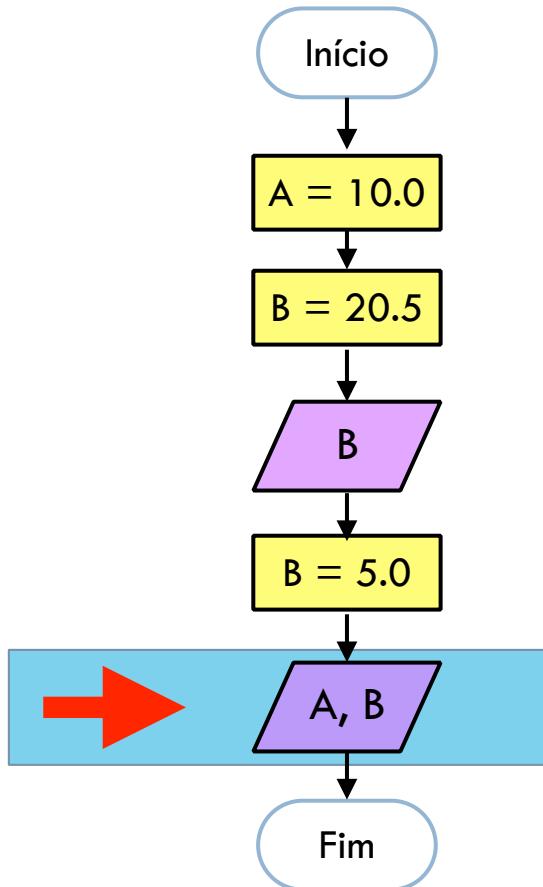
**Memória**



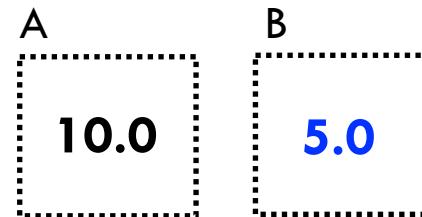
**O que o usuário ve?**

>> 20.5

# Exemplo



**Memória**



**O que o usuário ve?**

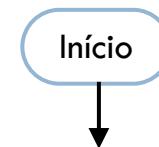
```
>> 20.5  
>> 10.0 5.0
```

## Exemplo 2

- Calcular a área de um círculo e comprimento da circunferência

# Exemplo 2

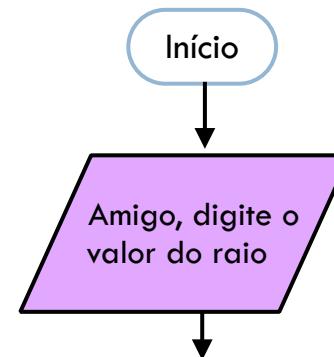
- Calcular a área de um círculo e comprimento da circunferência



# Exemplo 2

- Calcular a área de um círculo e comprimento da circunferência

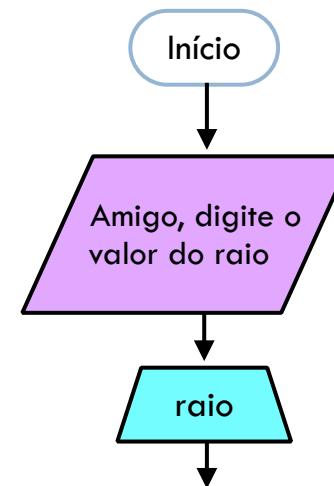
> Obter o valor do raio do círculo  
(Entrada)



# Exemplo 2

- Calcular a área de um círculo e comprimento da circunferência

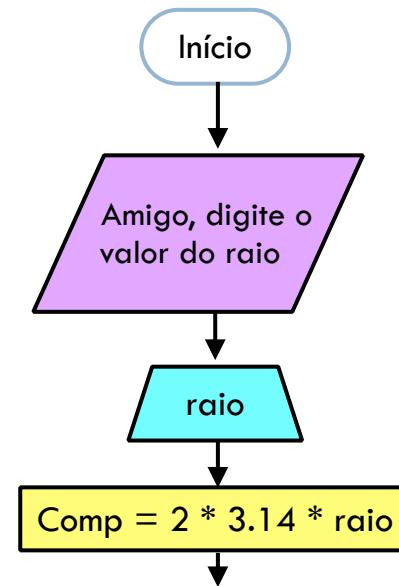
> Obter o valor do raio do círculo  
(Entrada)



# Exemplo 2

- Calcular a área de um círculo e comprimento da circunferência

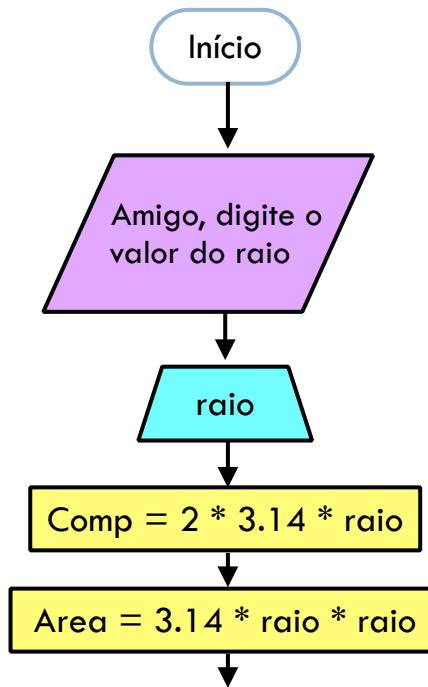
> Obter o valor do raio do círculo  
(Entrada)  
> Calcular a área



# Exemplo 2

- Calcular a área de um círculo e comprimento da circunferência

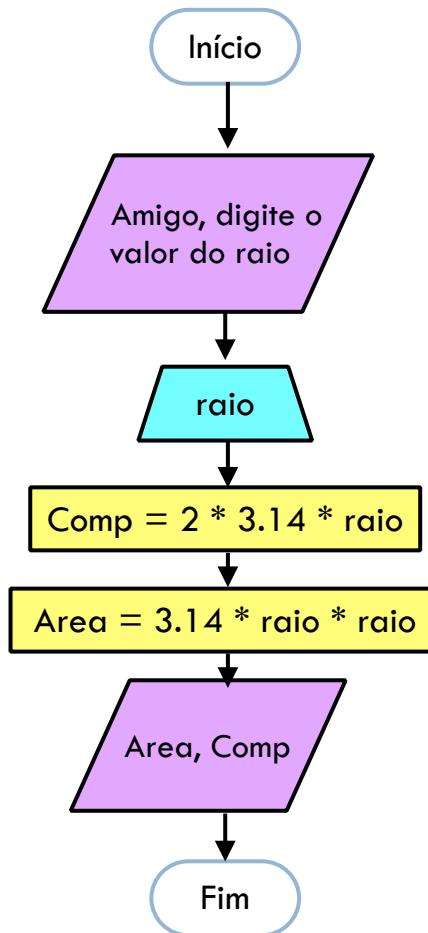
> Obter o valor do raio do círculo  
(Entrada)  
> Calcular a área  
> Calcular o comprimento



# Exemplo 2

- Calcular a área de um círculo e comprimento da circunferência

> Obter o valor do raio do círculo  
(Entrada)  
> Calcular a área  
> Calcular o comprimento  
> Mostrar os valores de área e  
comprimento calculados (Saída)



# Exercícios

1

O que será impresso na tela para o usuário em cada programa?

a)

```
A = 30  
B = 20  
C = A + B  
Imprimir C  
B = 10  
Imprimir B, C  
C = A + B  
Imprimir A, B, C
```

b)

```
A = 10  
B = 20  
C = A  
B = C  
A = B  
Imprimir A, B, C
```

c)

```
A = 10  
B = A + 1  
A = B + 1  
B = A + 1  
Imprimir A  
A = B + 1  
Imprimir A, B
```

# Roteiro

- 1 Introdução**
- 2 Fluxograma**
- 3 Variáveis**
- 4 Operadores e Estruturas de Controle**
- 5 Exercícios**
- 6 Referências**

# Operadores Aritméticos



Operador	Operação
( )	Parênteses
	Exponenciação
%	Módulo
/	Divisão
*	Multiplicação
+	Soma
-	Subtração

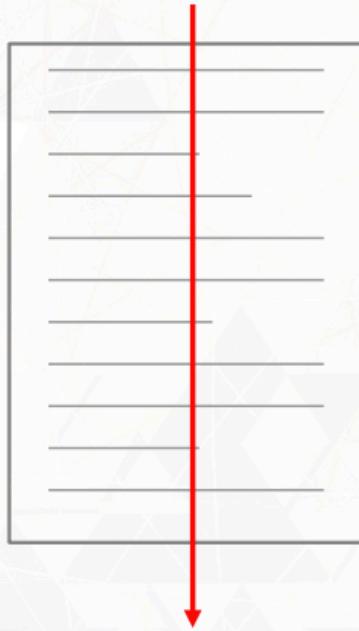
O operador % retorna o resto da divisão entre dois operandos

$4 \% 2 = 0$      $5 \% 2 = 1$      $17 \% 3 = 2$      $100 \% 50 = 0$

# Estrutura de Controle

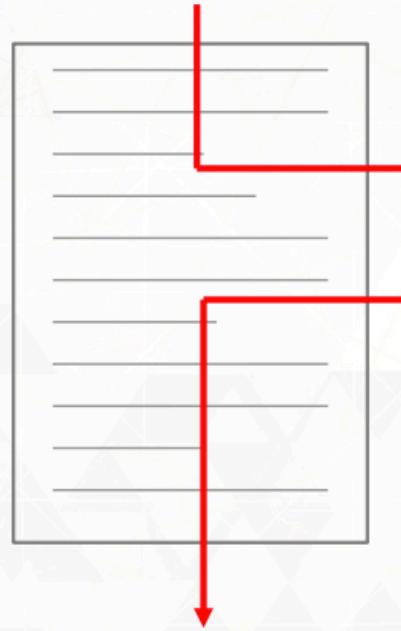
- Programa de computador
  - conjunto de instruções organizadas de forma a produzir a solução de um determinado problema
- Fluxo de Execução
  - Começa na primeira linha e avança sequencialmente
    - de cima para baixo
  - Em muitas circunstâncias é necessário executar instruções em um ordem diferente
  - Necessidade de decisão entre fluxos alternativos de execução ou da repetição de determinadas instruções
  - Pode haver bifurcações, repetição de código e tomada de decisão

# Estrutura de Controle



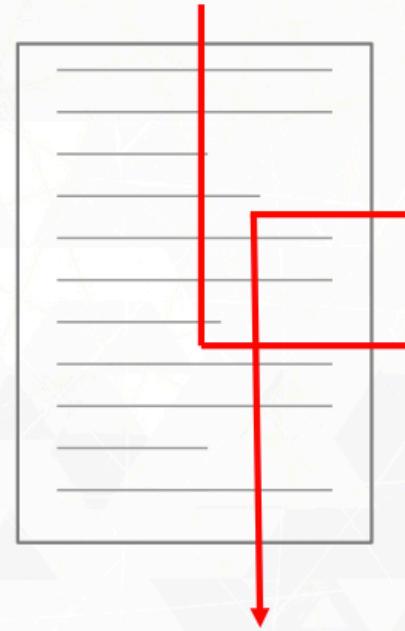
**Fluxo de execução sequencial**

*Comandos são executados um após o outro*



**Fluxo de execução com desvio**

*Comandos são executados dependendo do valor de uma condição*



**Fluxo de execução repetitivo**

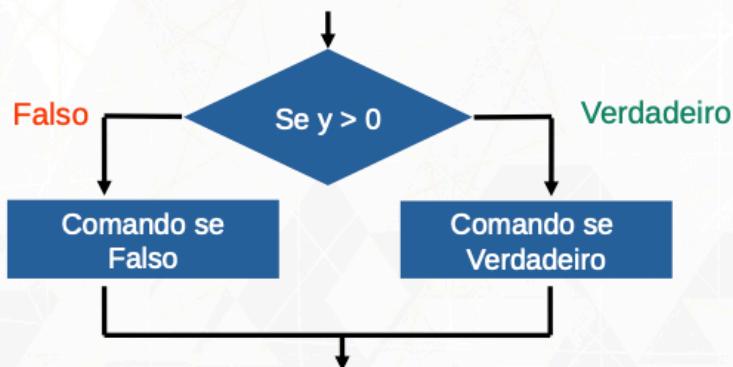
*Comandos são executados de forma repetida*

# Estrutura de Controle

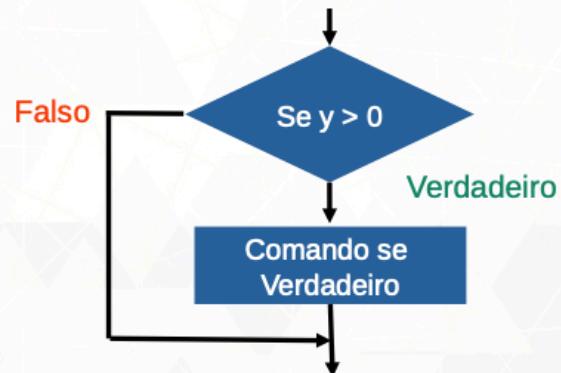
- As estruturas de controle dividem-se em:
  - Estruturas de seleção
  - Estruturas de repetição (*loops*)
- As estruturas de controle estão vinculadas às condições que determinam se instruções serão ou não executadas
- Uma condição de controle está relacionada aos operadores relacionais e lógicos

# Estrutura de Controle

## Seleção



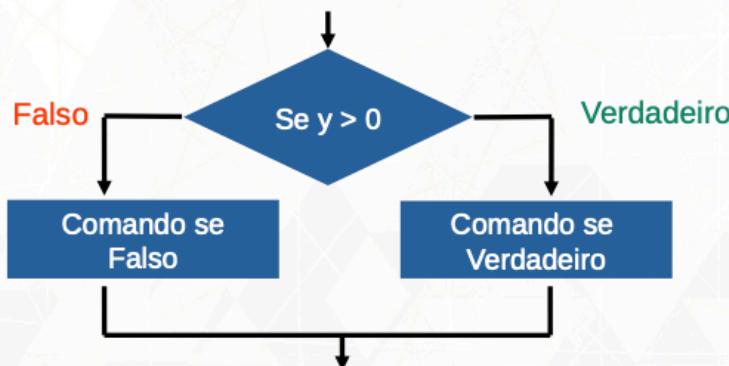
**Se  $y$  é maior que zero**  
Executa Comando se **Verdadeiro**  
**Senão**  
Executa Comando se **Falso**



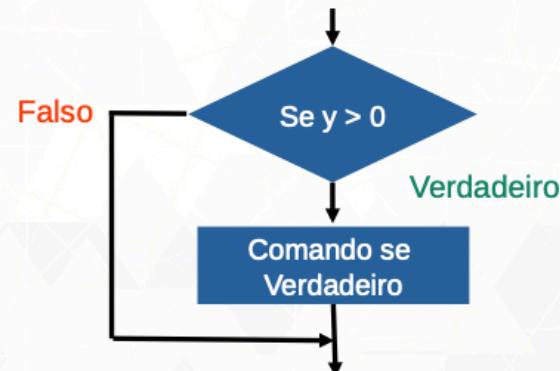
**Se  $y$  é maior que zero**  
Executa Comando se **Verdadeiro**

# Estrutura de Controle

## Seleção



**Se  $y$  é maior que zero**  
Executa Comando se **Verdadeiro**  
**Senão**  
Executa Comando se **Falso**



**Se  $y$  é maior que zero**  
Executa Comando se **Verdadeiro**

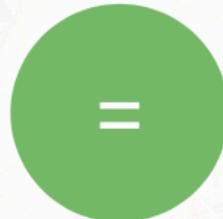
**NUNCA** executa o verdadeiro e falso simultaneamente

# Operadores relacionais

- Operadores relacionais comparam dois valores e retornam um valor booleano (**verdadeiro** ou **falso**)
- Usados em estruturas condicionais e laços de repetição

Operador	Descrição	X	Y	Lógico	Resultado
<code>==</code>	Igual a	2	3	<code>X == Y</code>	Falso
<code>!=</code>	Diferente de	2	3	<code>X != Y</code>	Verdadeiro
<code>&gt;</code>	Maior que	2	3	<code>X &gt; Y</code>	Falso
<code>&gt;=</code>	Maior ou igual	2	3	<code>X &gt;= Y</code>	Falso
<code>&lt;</code>	Menor que	2	3	<code>X &lt; Y</code>	Verdadeiro
<code>&lt;=</code>	Menor ou igual	2	3	<code>X &lt;= Y</code>	Verdadeiro

# Operador de atribuição



Operador de atribuição

$X = 5$

X recebe o valor 5



Operador relacional

$X == 5$

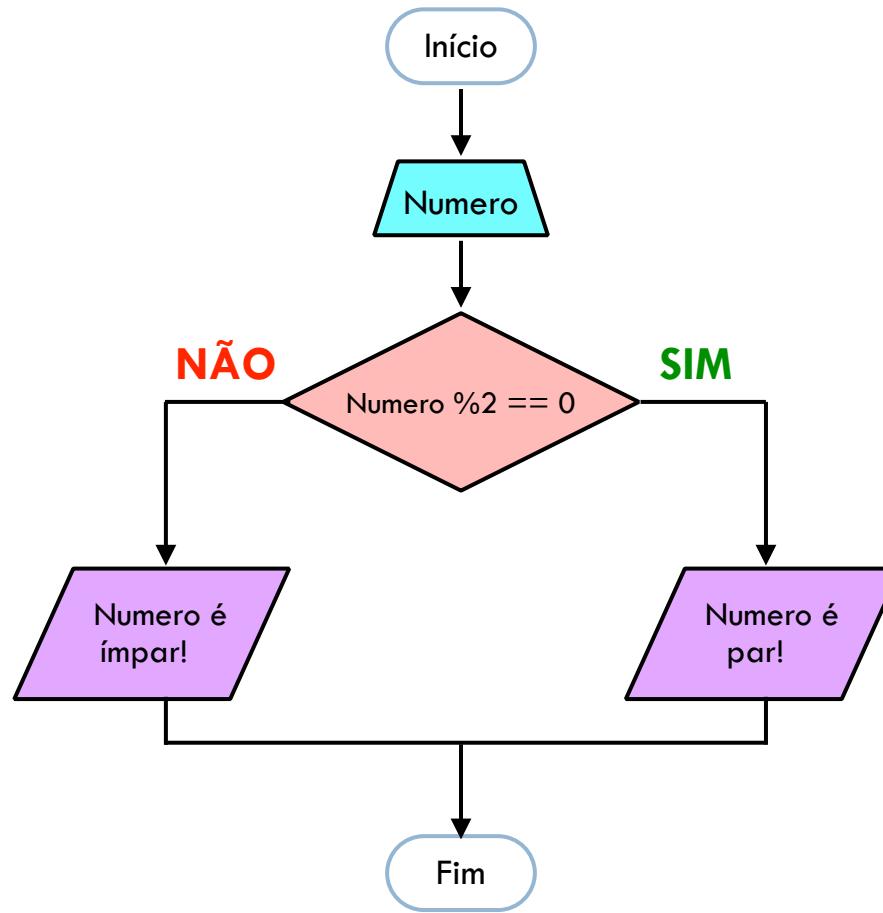
O valor armazenado em X  
é igual a 5?

# Exemplo

- Determinar se um número é par ou ímpar?

# Exemplo

- Determinar se um número é par ou ímpar?



# Roteiro

- 1 Introdução**
- 2 Fluxograma**
- 3 Variáveis**
- 4 Operadores e Estruturas de Controle**
- 5 Exercícios**
- 6 Referências**

# Exercícios

- 2** Faça um programa que leia dois números quaisquer e imprima o maior deles
- 3** Faça um programa receba duas notas de um aluno, calcule e mostre a média aritmética das notas e uma mensagem conforme a tabela a seguir:

Media	Mensagem
De 0.0 a 3.9	Reprovado
De 4.0 a 5.9	Exame
De 6.0 a 10.0	Aprovado

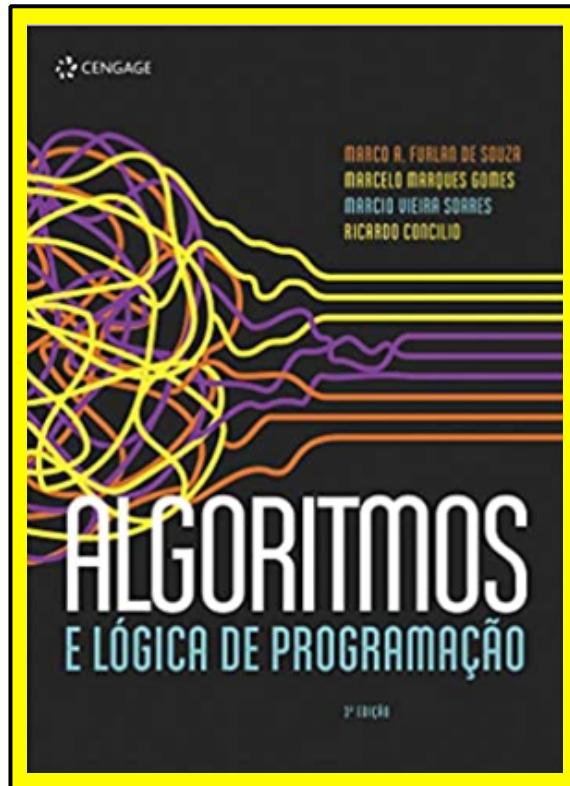
# Exercícios

- 4** Faça um programa que leia dois números quaisquer e imprima o resultado do cálculo do maior dividido pelo menor
- 5** Refaça o exercício anterior, mas agora imprima uma mensagem de erro caso puder ocorrer uma divisão por zero.
- 6** Desenvolver a lógica para um programa que efetue o cálculo do reajuste de salário de um funcionário
  - salário ≤ 500, reajuste de 15%
  - salário > 500 mas salário ≤ 1000, reajuste de 10%
  - salário > 1000, reajuste será de 5%
- 7** Faça um algoritmo que leia os valores A, B e C, e imprima o resultado somente se a soma de A + B é menor que C. Caso não seja, não faça nada e encerre o algoritmo.

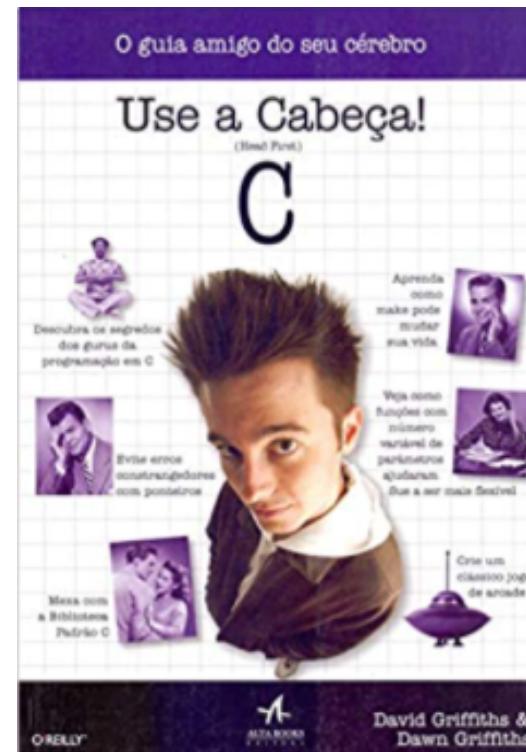
# Roteiro

- 1 Introdução**
- 2 Fluxograma**
- 3 Variáveis**
- 4 Operadores e Estruturas de Controle**
- 5 Exercícios**
- 6 Referências**

# Referências sugeridas



[Souza et al, 2019]



[Griffiths & Griffiths, 2013]

# Perguntas?

Prof. Rafael G. Mantovani

[rafaelmantovani@utfpr.edu.br](mailto:rafaelmantovani@utfpr.edu.br)