

# Algoritmos e Programação de Computadores



Ricardo José Cabeça de Souza

[ricardo.souza@ifpa.com.br](mailto:ricardo.souza@ifpa.com.br)

# SUMÁRIO



- ▶ Estrutura de um algoritmo
- ▶ Estrutura de Seleção

# OBJETIVOS

- ▶ Apresentar a estrutura básica de um algoritmo
- ▶ Analisar um exemplo de algoritmo usando **SELEÇÃO**

# PSEUDOLINGUAGEM

## ▶ ESTRUTURA DE UM ALGORITMO

Início

[Declaração de Variáveis/Constantes];

Comandos;

...

...

Fim.

[ ] Representa que pode existir ou não (Depende do problema)

## ▶ ESTRUTURA DE UM ALGORITMO COM SELEÇÃO

Início

Declaração de Variáveis/Constantes;

....

Se (TESTE\_LÓGICO)

então

Comando(s) caso resultado TESTE\_LÓGICO seja VERDADEIRO;

...

senão

Comando(s) caso resultado TESTE\_LÓGICO seja FALSO;

...

Fim\_se;

...

Fim.

# RESOLUÇÃO DO PROBLEMA

## ▶ EXEMPLO 1

- Problema:
  - Fazer um algoritmo para ler dois números inteiros, verificar e informar se os números são iguais ou diferentes
- DEFINIÇÃO DO PROBLEMA
  - Está claro?
  - Alguma dúvida?

# RESOLUÇÃO DE PROBLEMAS

## ▶ ANÁLISE DO PROBLEMA

### ◦ ENTRADA

Representação da(s) Entrada(s):

Quais as informações precisam ser informadas para ser possível resolver o problema?

**Os dois números**

Representação dos Números: **X e Y**

# RESOLUÇÃO DE PROBLEMAS

## ► ANÁLISE DO PROBLEMA

- COMO RESOLVER?

Representação da Operação:

Realização de **Teste Lógico** entre os dois números informados (X e Y)

Se (**X = Y**)

então (Teste Lógico VERDADEIRO)

**Informar que os números são iguais**

senão (Teste Lógico FALSO)

**Informar que os números são diferentes**



# RESOLUÇÃO DE PROBLEMAS

## ▶ ANÁLISE DO PROBLEMA

- SAÍDA

Representação da Saída

**A saída é o próprio resultado do Teste Lógico entre os dois números informados (X e Y)**

**Se (X = Y)**  
**então (Teste Lógico VERDADEIRO)**  
**Informar que os números são iguais**  
**senão (Teste Lógico FALSO)**  
**Informar que os números são diferentes**

# RESOLUÇÃO DE PROBLEMAS

## ► Sequência de operações para execução algoritmo:

- a) Realizar a entrada do primeiro valor (X)
- b) Realizar a entrada do segundo valor (Y)
- c) Realizar o teste e informar o resultado:

Se  $(X = Y)$

então

Informar que os números são iguais

senão

Informar que os números são diferentes

## ▶ ESTRUTURA DE UM ALGORITMO COM SELEÇÃO

Início

Declaração de Variáveis/Constantes;

....

Se (TESTE\_LÓGICO)

então

Comando(s) caso resultado TESTE\_LÓGICO seja VERDADEIRO;

...

senão

Comando(s) caso resultado TESTE\_LÓGICO seja FALSO;

...

Fim\_se;

...

Fim.

# PSEUDOLINGUAGEM



## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)

Início

Inteiro: X,Y;

Imprimir("Digite o primeiro número inteiro:");

Ler(X);

Imprimir("Digite o segundo número inteiro:");

Ler(Y);

Se (X = Y)

então

Imprimir("Os números digitados são iguais!");

senão

Imprimir("Os números digitados são diferentes!");

Fim\_se;

Fim.

# PSEUDOLINGUAGEM



## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)

### ◦ INDENTAÇÃO

- Em linguagem de programação é a digitação dos códigos do programa, afastados por espaço da margem e dispostos hierarquicamente
- Objetivo: para facilitar a visualização e percepção do programa
  - Na linguagem Python usada para estruturar blocos

# PSEUDOLINGUAGEM



## ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)

Início

Inteiro: X,Y;

Imprimir("Digite o primeiro número inteiro:");

Ler(X);

Imprimir("Digite o segundo número inteiro:");

Ler(Y);

Se (X = Y)

então

Imprimir("Os números digitados são iguais!");

senão

Imprimir("Os números digitados são diferentes!");

Fim\_se;

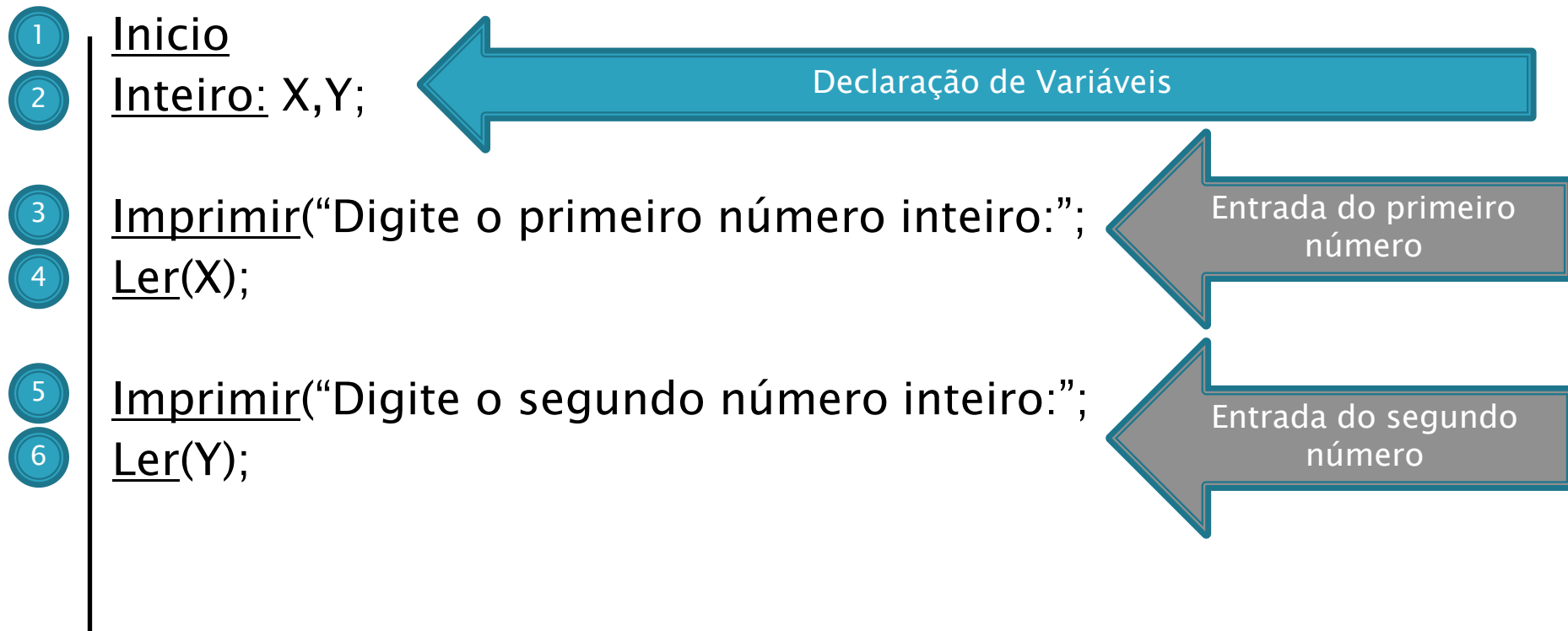
Fim.

**Bloco de Comandos**

# PSEUDOLINGUAGEM

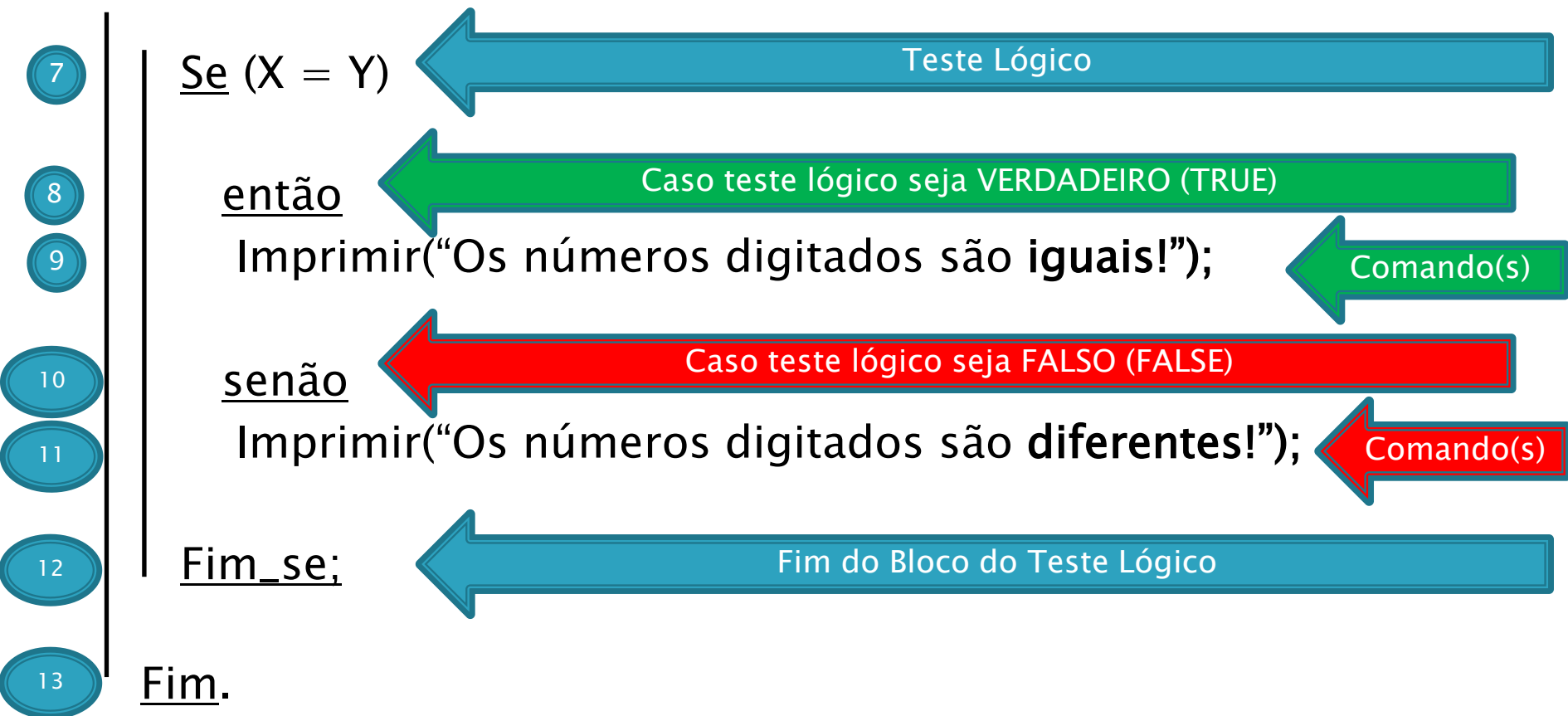


## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)



# PSEUDOLINGUAGEM

## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)





# PSEUDOLINGUAGEM

## ▶ TESTE DE MESA

- Após o desenvolvimento de um algoritmo, é necessário verificar cada um dos passos que foram definidos, ou seja, efetuar um **teste**
- Para isso, devemos ler cada uma das instruções e anote o resultado de cada tarefa/passo, verificando possíveis erros ou outras formas de solucionar o problema
- É um processo **manual** que é utilizado para validar a lógica de um determinado algoritmo

# PSEUDOLINGUAGEM

## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)



**MEMÓRIA RAM SIMULADA**

2

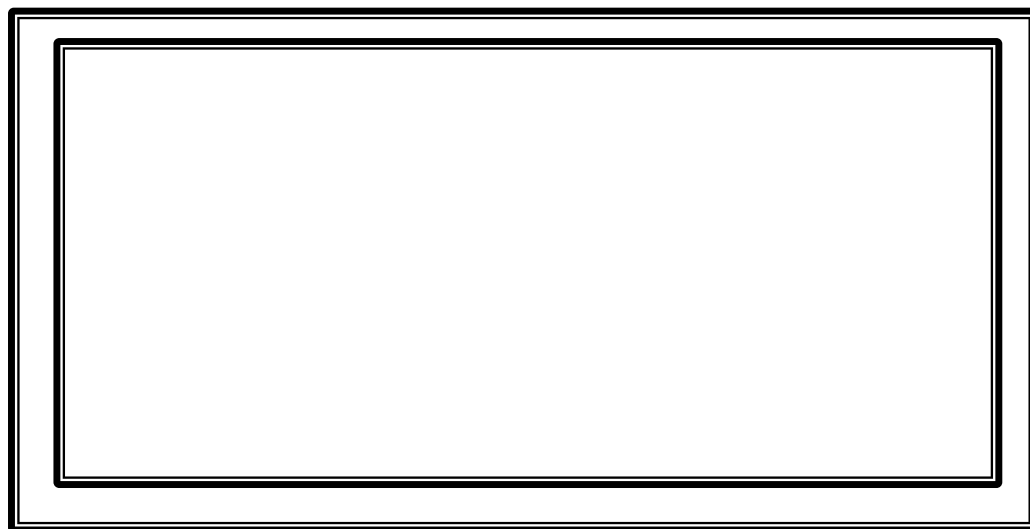


X



Y

**MONITOR COMPUTADOR SIMULADO**



# PSEUDOLINGUAGEM

## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)

3

Imprimir("Digite o primeiro número inteiro:");  
Ler(X);

Entrada do primeiro  
número

Aguarda usuário digitar  
valor

**MEMÓRIA RAM SIMULADA**



X



Y

**MONITOR COMPUTADOR SIMULADO**

3

Digite o primeiro número inteiro: \_

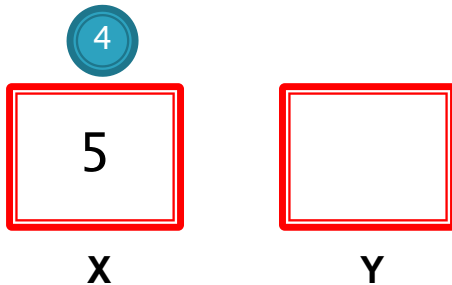
# PSEUDOLINGUAGEM

## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)

3 | Imprimir("Digite o primeiro número inteiro:");  
4 | Ler(X);

Entrada do primeiro  
número

### MEMÓRIA RAM SIMULADA



### MONITOR COMPUTADOR SIMULADO

3

Digite o primeiro número inteiro: 5

Usuário digita valor e  
tecla ENTER (4)

# PSEUDOLINGUAGEM

## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)

5

Imprimir("Digite o segundo número inteiro:");  
Ler(Y);

Entrada do segundo  
número

### MEMÓRIA RAM SIMULADA

5

X

Y

### MONITOR COMPUTADOR SIMULADO

5

Digite o primeiro número inteiro: 5  
Digite o segundo número inteiro: \_

Aguarda usuário  
digitar valor

# PSEUDOLINGUAGEM

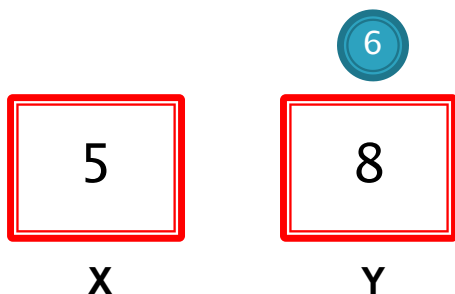
## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)

5 | Imprimir("Digite o segundo número inteiro:");  
6 | Ler(Y);

Entrada do segundo  
número

Usuário digita  
valor e tecla ENTER  
(6)

### MEMÓRIA RAM SIMULADA

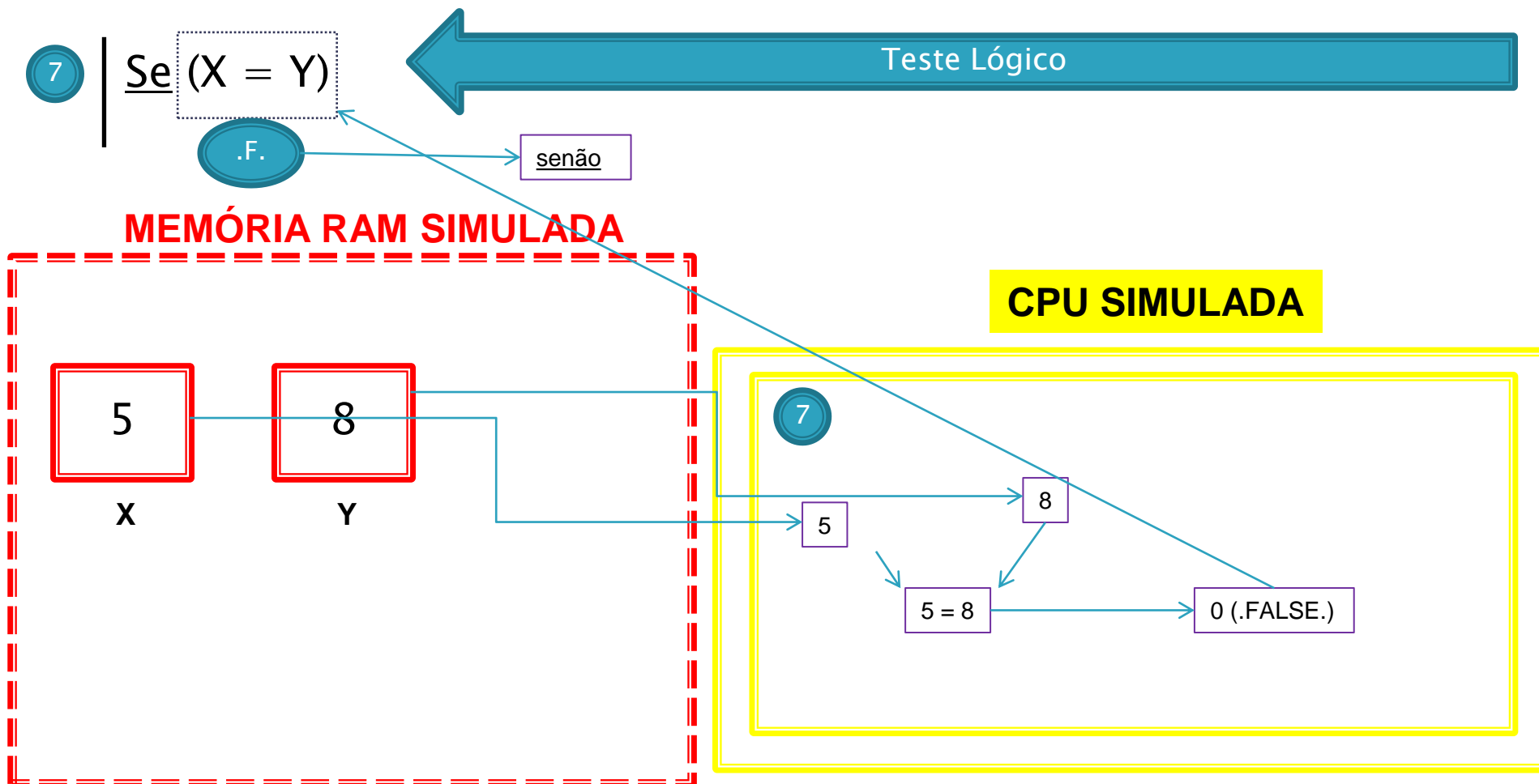


### MONITOR COMPUTADOR SIMULADO

5 | Digite o primeiro número inteiro: 5  
Digite o segundo número inteiro: 8

# PSEUDOLINGUAGEM

## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)

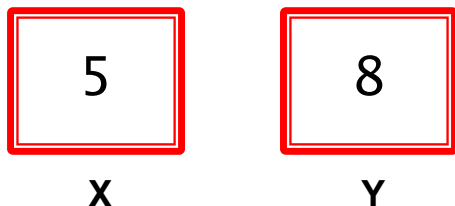


# PSEUDOLINGUAGEM

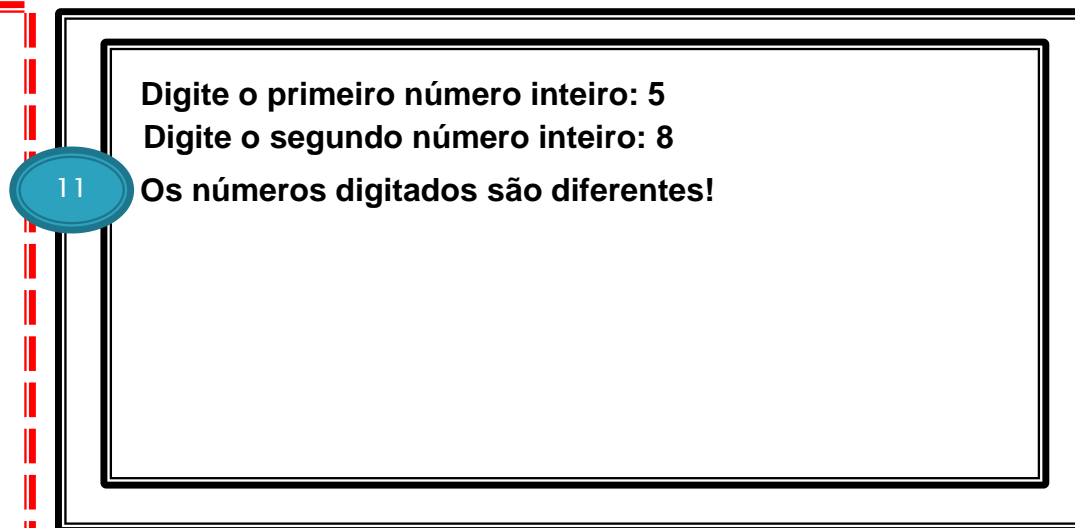
## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)



### MEMÓRIA RAM SIMULADA



### MONITOR COMPUTADOR SIMULADO





# RESOLUÇÃO DO PROBLEMA

## ▶ EXEMPLO 2

- Problema:
  - Fazer um algoritmo para verificar se um determinado número inteiro é par ou ímpar.
- DEFINIÇÃO DO PROBLEMA
  - Está claro?
  - Alguma dúvida?

# RESOLUÇÃO DE PROBLEMAS

## ▶ ANÁLISE DO PROBLEMA

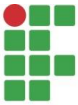
### ◦ ENTRADA

Representação da(s) Entrada(s):

Quais as informações precisam ser informadas para ser possível resolver o problema?

**Um número inteiro**

Representação do Número: **NUM**



# RESOLUÇÃO DE PROBLEMAS

## ► ANÁLISE DO PROBLEMA

### ◦ COMO RESOLVER?

**Análise da definição de número par:** um número é par se ao ser dividido pelo número dois (divisão inteira), tiver como resultado dessa divisão, um número inteiro.

Logo, ao realizar a **divisão inteira** de qualquer inteiro por **2**, será **PAR** se o **resto for 0 (zero)**.

Realização de **Teste Lógico** com base no **RESTO** de **NUM**:

Se ( **(NUM mod 2) = 0**)

então (Teste Lógico VERDADEIRO)

Informar que o número lido é **PAR**

senão (Teste Lógico FALSO)

Informar que o número lido é **ÍMPAR**

# RESOLUÇÃO DE PROBLEMAS

## ► ANÁLISE DO PROBLEMA

- SAÍDA

Representação da Saída

A saída é o próprio resultado do Teste Lógico com o RESTO

Se ( **(NUM mod 2) = 0** )  
então (Teste Lógico VERDADEIRO)  
    **Informar que o número lido é PAR**  
senão (Teste Lógico FALSO)  
    **Informar que o número lido é ÍMPAR**

# RESOLUÇÃO DE PROBLEMAS

- ▶ Sequência de operações para execução algoritmo:
  - a) Realizar a entrada do número inteiro (NUM)
  - b) Realizar o teste e informar o resultado:
    - Se (  $(\text{NUM} \bmod 2) = 0$  )
      - então (Teste Lógico VERDADEIRO)
      - Informar que o número lido é PAR
    - senão (Teste Lógico FALSO)
    - Informar que o número lido é ÍMPAR

## ▶ ESTRUTURA DE UM ALGORITMO COM SELEÇÃO

Início

Declaração de Variáveis/Constantes;

....

Se (TESTE\_LÓGICO)

então

Comando(s) caso resultado TESTE\_LÓGICO seja VERDADEIRO;

...

senão

Comando(s) caso resultado TESTE\_LÓGICO seja FALSO;

...

Fim\_se;

...

Fim.

# PSEUDOLINGUAGEM



## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)

Início

Inteiro: NUM;

Imprimir("Informe um número inteiro:");

Ler(NUM);

Se ( (NUM mod 2) = 0)

então

Imprimir(NUM, " é um número PAR!");

senão

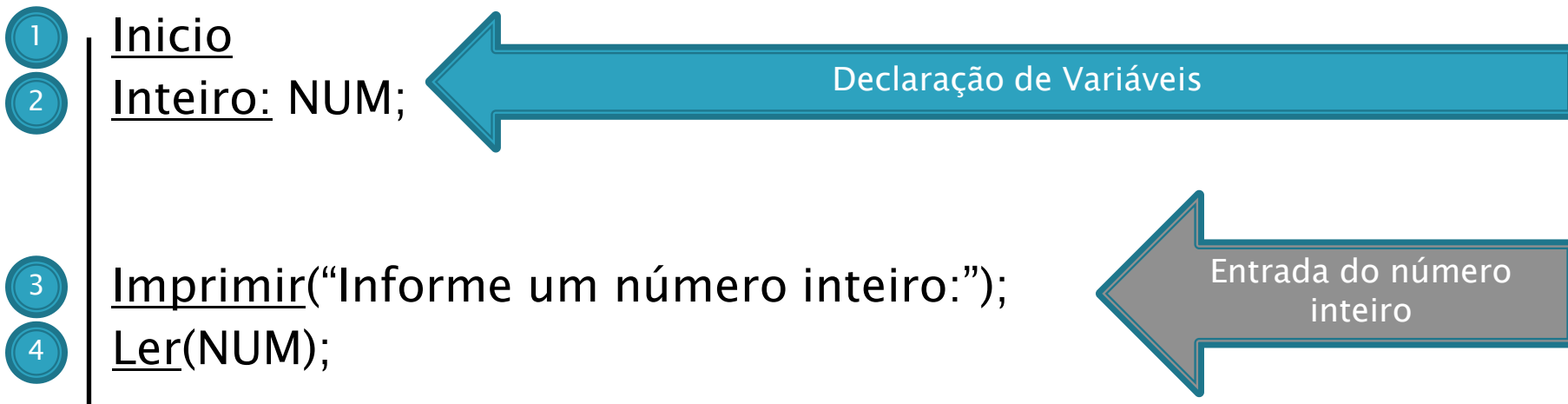
Imprimir(NUM, " é um número ÍMPAR!");

Fim\_se;

Fim.

# PSEUDOLINGUAGEM

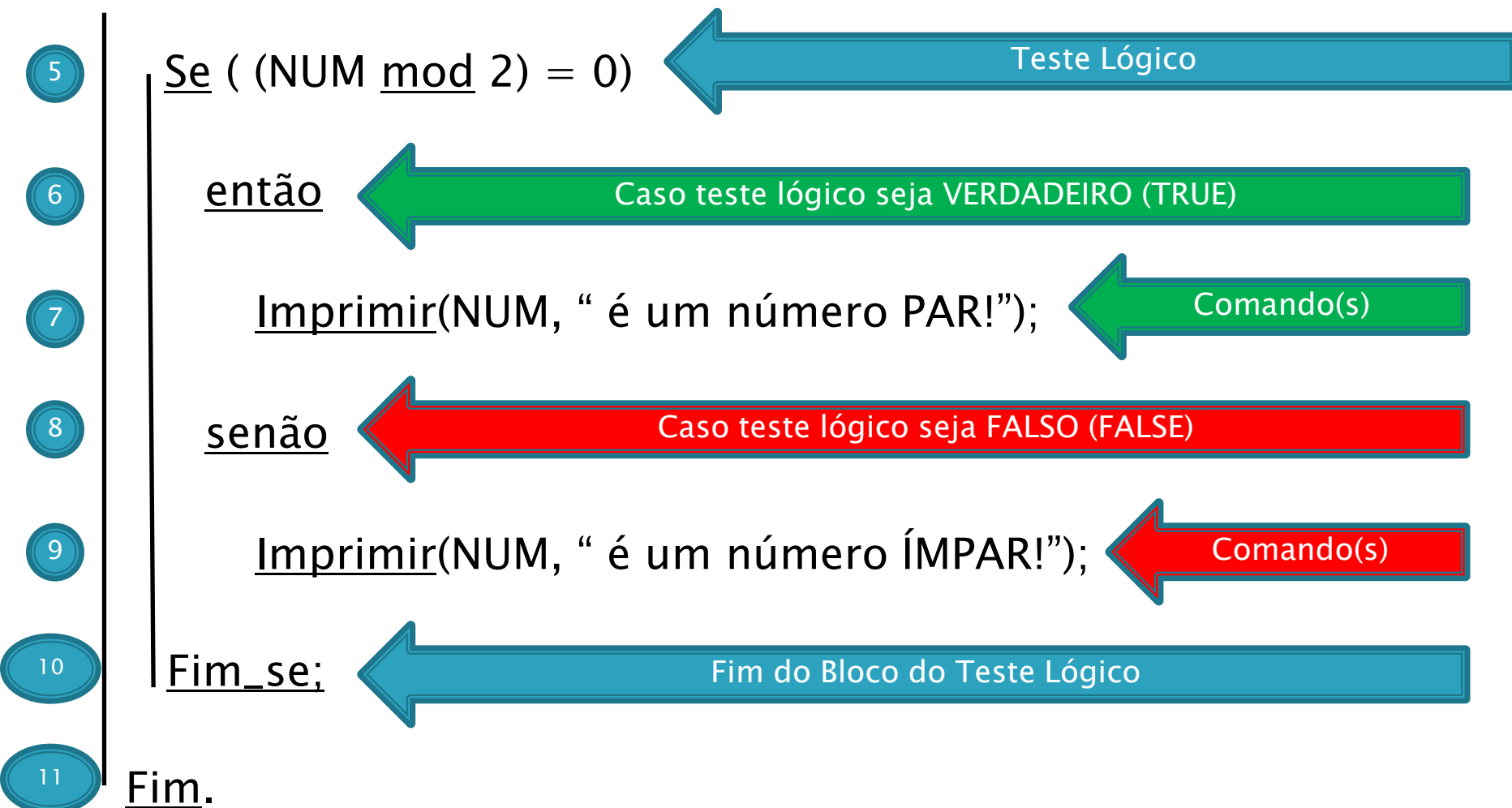
## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)





# PSEUDOLINGUAGEM

## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)



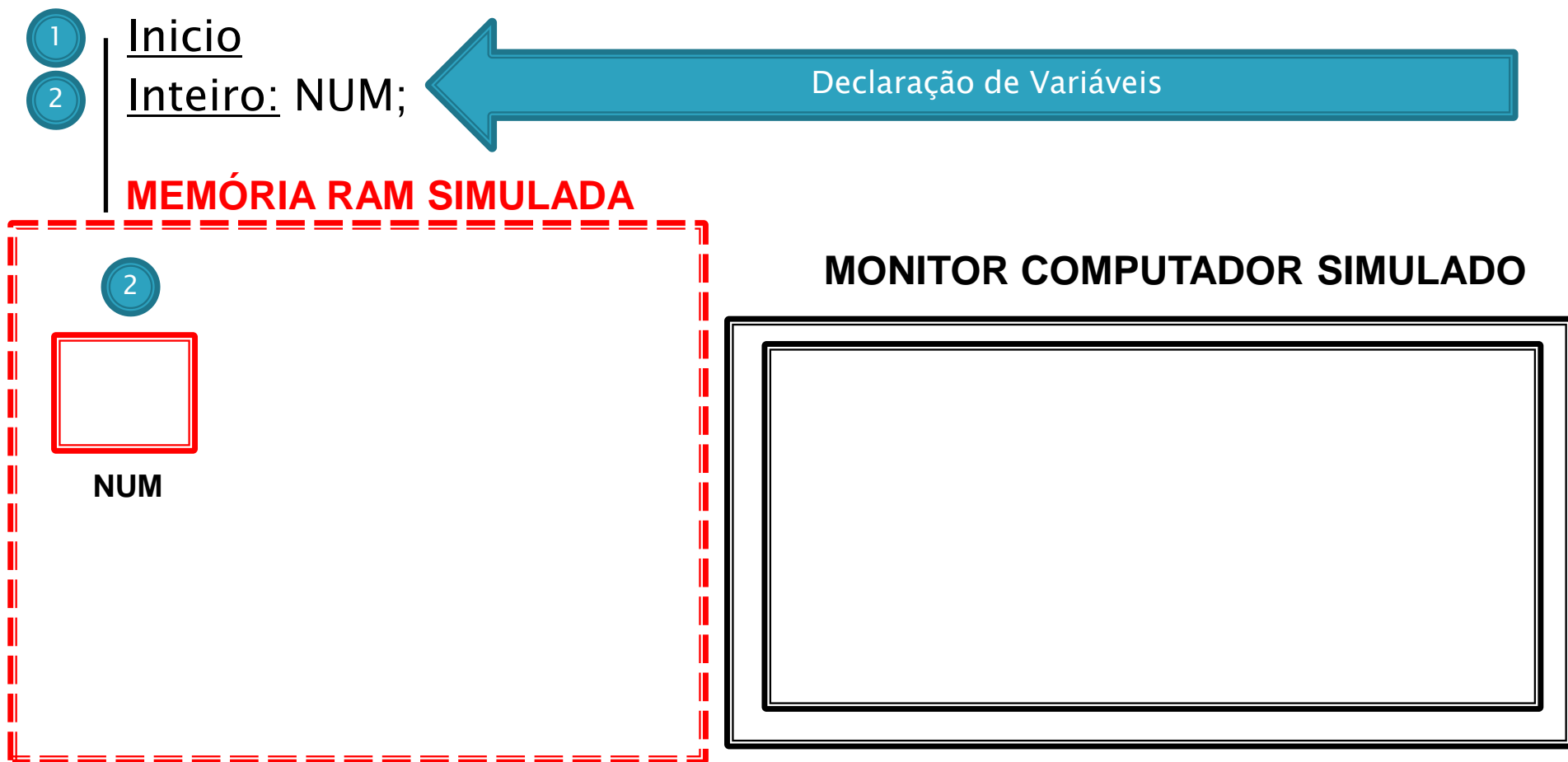
# PSEUDOLINGUAGEM

## ▶ TESTE DE MESA

- Após o desenvolvimento de um algoritmo, é necessário verificar cada um dos passos que foram definidos, ou seja, efetuar um **teste**
- Para isso, devemos ler cada uma das instruções e anote o resultado de cada tarefa/passo, verificando possíveis erros ou outras formas de solucionar o problema
- É um processo **manual** que é utilizado para validar a lógica de um determinado algoritmo

# PSEUDOLINGUAGEM

## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)



# PSEUDOLINGUAGEM

## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)

3

```
Imprimir("Informe um número inteiro:");  
Ler(NUM);
```

**MEMÓRIA RAM SIMULADA**



NUM

**MONITOR COMPUTADOR SIMULADO**

3

Informe um número inteiro: \_

Entrada número inteiro

Aguarda usuário digitar  
valor

# PSEUDOLINGUAGEM

## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)

3 | Imprimir("Informe um número inteiro:");  
4 | Ler(NUM);

### MEMÓRIA RAM SIMULADA

4  
6

NUM

### MONITOR COMPUTADOR SIMULADO

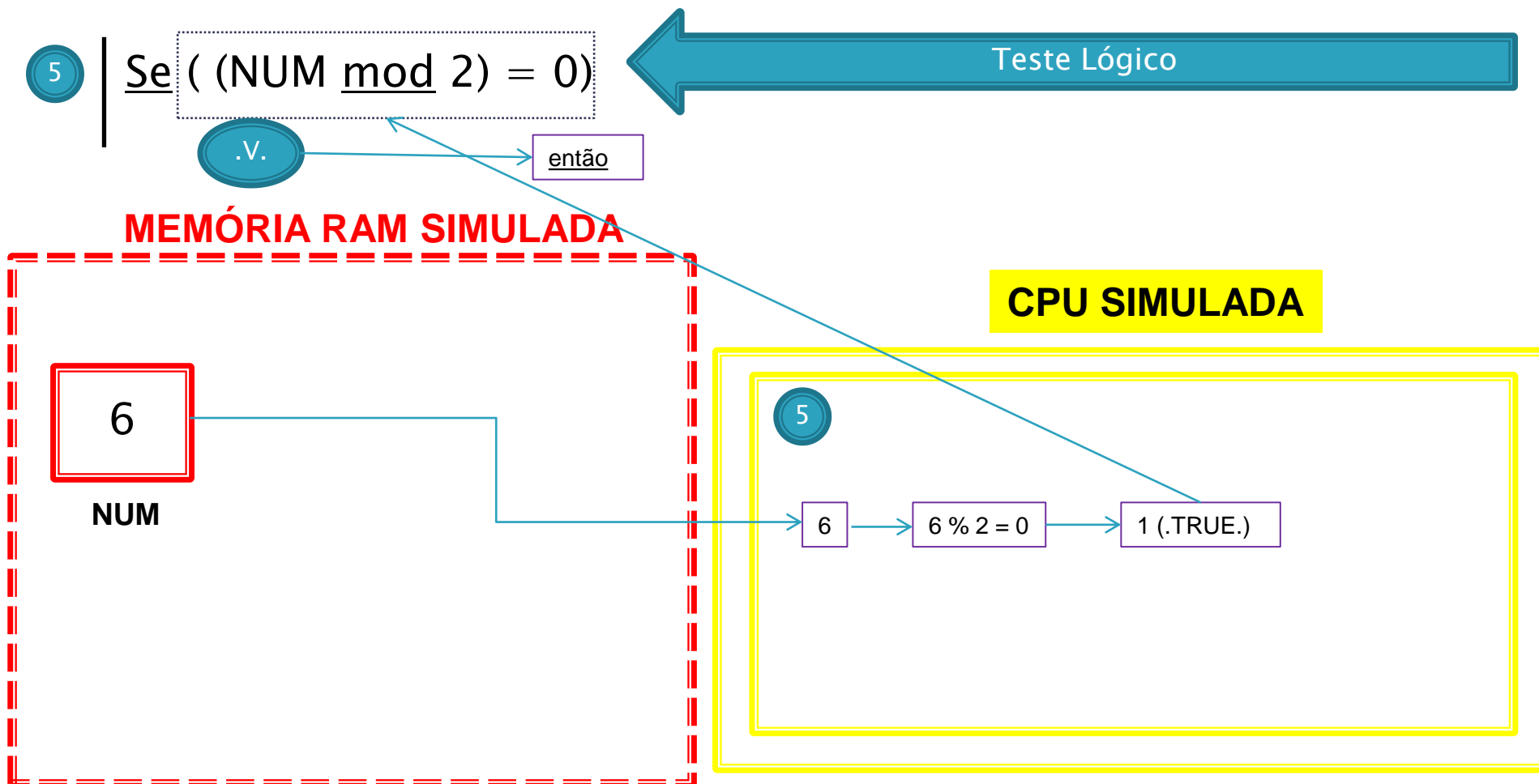
3 | Informe um número inteiro: 6

Entrada número inteiro

Usuário digita valor e  
tecla ENTER (4)

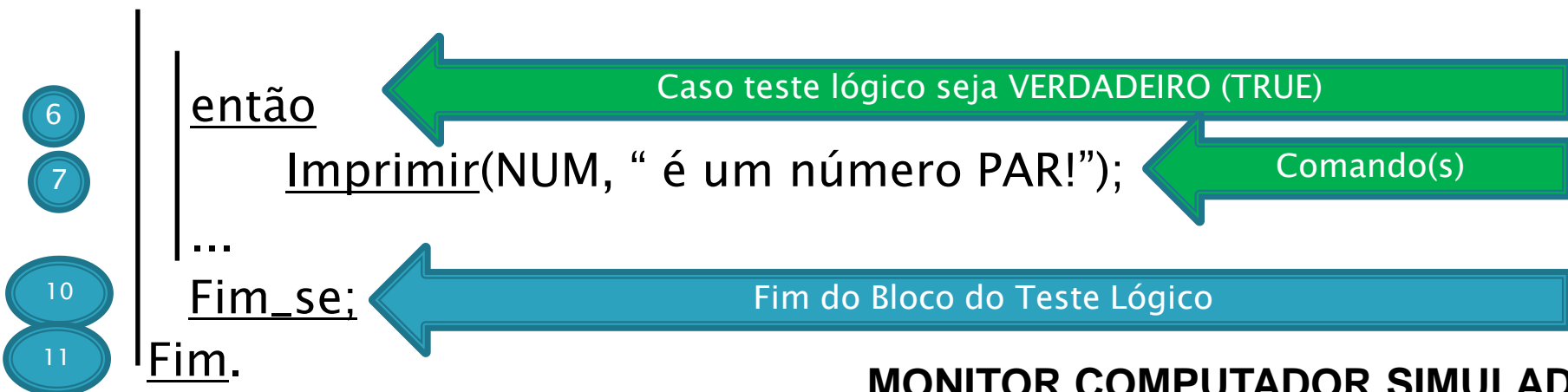
# PSEUDOLINGUAGEM

## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)



# PSEUDOLINGUAGEM

## ▶ ESTRUTURA DO ALGORITMO (PSEUDOLINGUAGEM)



### MEMÓRIA RAM SIMULADA

6

NUM

### MONITOR COMPUTADOR SIMULADO

Informe um número inteiro: 6

6 é um número PAR!

11



# REFERÊNCIAS

- ▶ MENEZES, Nilo Ney Coutinho. **Introdução à Programação com Python: algoritmos e lógica de programação para iniciantes**. 2. ed. São Paulo: Novatec Editora Ltda, 2014. 325 p.
- ▶ VELOSO, Paulo. et. al. **Estrutura de dados**. Rio de Janeiro: Campus, 1986.
- ▶ MORAES, Celso Roberto. **Estrutura de dados e algoritmos**. 2. ed. São Paulo: Futura, 2003.
- ▶ CELES, W. Rangel, J. L. **Curso de Estrutura de Dados**. PUC-Rio, 2002.
- ▶ W. Celes, R. Cerqueira, J.L. Rangel. **Introdução a Estruturas de Dados – com técnicas de programação em C**. Rio de Janeiro: Campus, 2004.
- ▶ SCHILDT, Herbert. **C Completo e total**. São Paulo: MAKRON BOOKS, 1997.
- ▶ LOUDON, Kyle. **Dominando algoritmos com C**. São Paulo: CIENCIA MODERNA COMPUTAÇÃO, 2000.
- ▶ JAMSA, Kris. **Programando em C/C++: a bíblia**. São Paulo: Makron Books, 2000.
- ▶ LOPES, Anita. **Introdução a programação: 500 algoritmos resolvidos**. Rio de Janeiro: Campus, 2002.
- ▶ GIMARÃES, Ângelo de Moura. LAGES, Newton Alberto de Castilho. **Algoritmos e estruturas de dados**. Rio de Janeiro: Campus, 1985.
- ▶ LAUREANO, Marcos. **Programando em C para Linux, Unix e Windows**. Rio de Janeiro: BRASPORT LIVROS, 2005.
- ▶ MEDINA, Marco. FERTIG, Cristina. **Algoritmos e programação: teoria e prática**. São Paulo: NOVATEC INFORMATICA, 2005.
- ▶ ARAÚJO, Everton Coimbra de. **Algoritmos: fundamentos e prática**. 2. ed. Ampl. e atual. Florianópolis: VisualBooks, 2005.
- ▶ FEOFIOFF, Paulo. **Projeto de Algoritmos em C**. Disponível em <http://www.ime.usp.br/~pf/algoritmos/aulas/lista.html> acesso em 12/07/2011.