

## Estruturas de Controle de Fluxo

### 3. Estrutura ou laço de repetição

Ao analisarmos o que ocorre nos laços de repetição estudados (enquanto e repita) perceberemos que, normalmente, ocorre uma inicialização de uma variável, envolvida na expressão lógica que controla o número de repetições, e dentro do laço ocorre uma atualização no valor da variável mencionada, fazendo com que esta venha a tornar o resultado da avaliação da expressão lógica coerente para a finalização da execução do laço de repetição.

Com base nesta observação foi criado o laço de repetição **para**.

## Estruturas de Controle de Fluxo

### 3. Laços de repetição

Sintaxe:

...

para *<variável>* de *<valor-inicial>* ate *<valor-limite>*  
[*passo <incremento>*] faça

*<sequência-de-comandos>*

fimpara

...

*<variável >* É a variável contadora do número de repetições do laço (deve ser necessariamente uma variável do tipo inteiro)

## Estruturas de Controle de Fluxo

*<valor-inicial>* É uma expressão que especifica o valor de inicialização da variável contadora.

*<valor-limite>* É uma expressão que especifica o valor máximo ou mínimo que a variável contadora pode alcançar.

*<incremento>* É opcional. Quando presente, é precedido pela palavra-reservada *passo*, constitui-se de uma expressão que especifica o valor do incremento que será acrescentado à variável contadora em cada repetição do laço. O valor padrão, assumido por omissão, de *<incremento>* é 1. É possível especificar valores negativos para *<incremento>*.

**fimpara** Indica o fim da sequência de comandos a serem repetidos. Cada vez que o programa chega neste ponto, é acrescentado à variável contadora o valor de *<incremento>*, e o valor resultante é comparado a *<valor-limite>*. Se for menor ou igual (ou maior ou igual, quando *<incremento >* for negativo), a sequência de comandos será executada mais uma vez; caso contrário, a execução prosseguirá a partir do primeiro comando que esteja após o fimpara.

*<valor-inicial>*, *<valor-limite>* e *<incremento>* são avaliados uma **única vez** antes da execução da primeira repetição, e **não se alteram durante a execução do laço**, mesmo que variáveis eventualmente presentes nessas expressões tenham seus valores alterados.

# Estruturas de Controle de Fluxo

## 3. Laços de repetição (continuação)

### Exemplo 7:

O pseudocódigo e o fluxograma a seguir representam algoritmos que escrevem na saída padrão os números inteiros contidos no intervalo  $[1, 10]$ .

## Estruturas de Controle de Fluxo

algoritmo "exemplo 7"

var

valor: inteiro

inicio

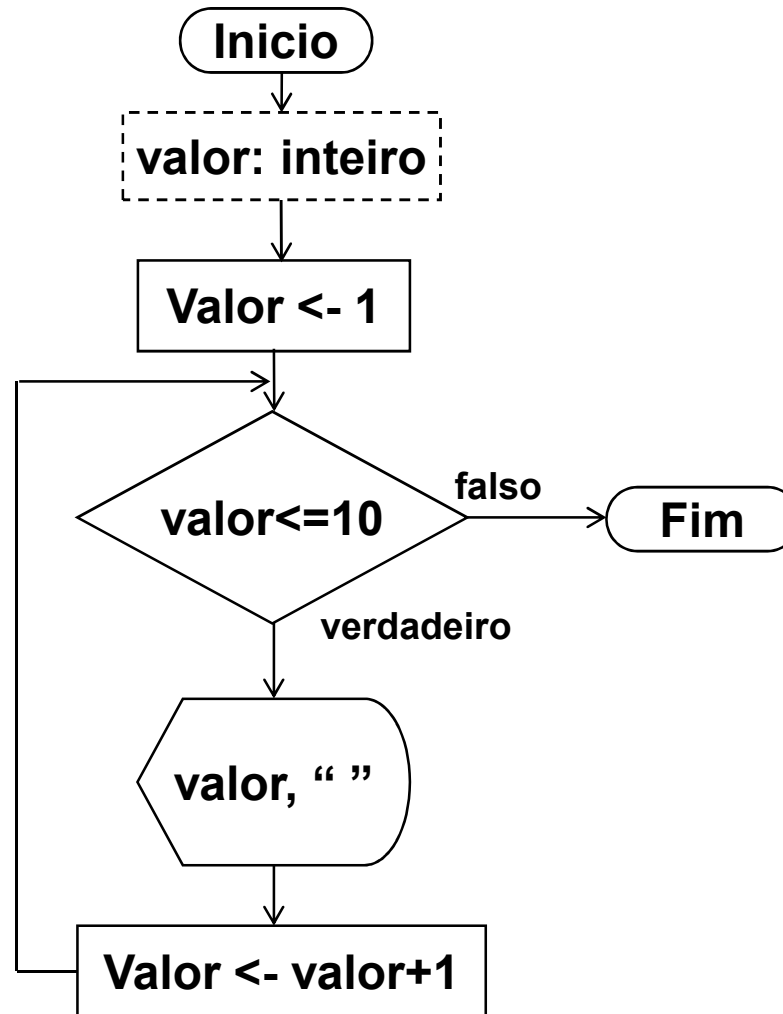
para valor de 1 ate 10 faca

escreval (valor)

fimpara

fimalgoritmo

# Estruturas de Controle de Fluxo



## Estruturas de Controle de Fluxo

### 3. Laços de repetição (continuação)

Exemplo 8:

Dada uma reta  $ax+by+c=0$  e cinco pontos, faça um algoritmo para calcular, para cada ponto, o seguinte: se o ponto estiver no primeiro quadrante calcule e informe a distância do ponto a reta caso contrário escreva uma mensagem informando que o ponto não pertence ao primeiro quadrante.



## Estruturas de Controle de Fluxo

algoritmo "exemplo 8"

var a,b,c,x,y: real

contador: inteiro

inicio

escreval ("Equação da reta:  $ax+by+c=0$ ")

escreva ("Coeficiente a da reta = ")

leia (a)

escreva ("Coeficiente b da reta = ")

leia (b)

escreva ("Coeficiente c da reta = ")

leia (c)

```

para contador de 1 ate 5 faca
    escreval ("Coordenadas do ponto ",contador, " :")
    escreva ("Coordenada x do ponto = ")
    leia (x)
    escreva ("Coordenada y do ponto = ")
    leia (y)
    se (x>=0 e y>=0) entao
        escreval ("A distância do ponto ",contador,
" a reta é: ",  $((a*x+b*y+c)^2)^{0.5}/(((a^2)+(b^2))^{0.5})$ )
    senao
        escreval ("O ponto ",contador, " não está no primeiro
quadrante! ")
    fimse
fimpara
fimalgoritmo

```

# Estruturas de Controle de Fluxo

## 3. Laços de repetição (continuação)

Exercício 20:

Construa um pseudocódigo para representar um algoritmo que exiba na saída padrão uma contagem decrescente do valor 30 até o valor 1.

## Estruturas de Controle de Fluxo

algoritmo "exercício 20"

var

  valor: inteiro

inicio

  para valor de 30 ate 1 passo -1 faça

    escreval (valor)

  fimpara

fimalgoritmo

## Estruturas de Controle de Fluxo

algoritmo "exercício 20"

var

valor: inteiro

inicio

para valor de 0 ate 29 faca

escreval (30-valor)

fimpara

fimalgoritmo

## Estruturas de Controle de Fluxo

### 3. Laços de repetição (continuação)

#### Exercício 21:

Construa um algoritmo, representando-o através de um pseudocódigo e de um fluxograma, que leia da entrada padrão um número inteiro positivo e retorne na saída padrão uma mensagem indicando se o número lido é ou não um número primo. As entradas devem ser validadas.

**algoritmo "exercício 21"**

**var A, i: inteiro**

**eh\_primo: logico**

**inicio**

**repita**

**escreva ("Entre com um valor inteiro positivo: ")**

**leia (A)**

**ate (A>0)**

**eh\_primo <- verdadeiro**

**para i de 1+1 ate A-1 faça**

**se (A%i=0) entao**

**eh\_primo <- falso**

**fimse**

**fimpara**

**se (eh\_primo e A<>1) entao**

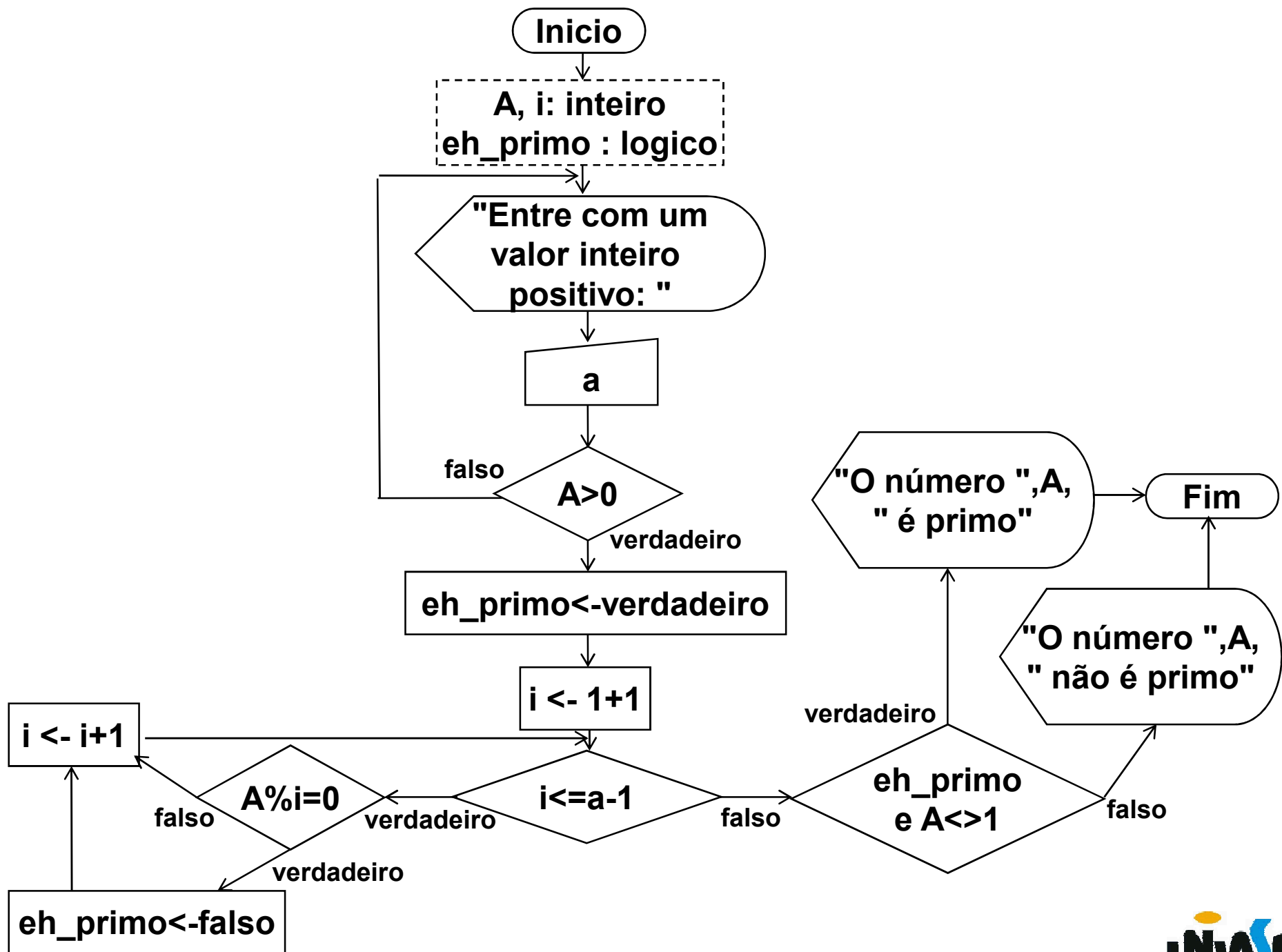
**escreva ("O número ",A," é primo")**

**senao**

**escreva ("O número ",A," não é primo")**

**fimse**

**174 fimalgoritmo**





**algoritmo "exercício 21 usando interrompa"**

**var A, i: inteiro**

**eh\_primo: logico**

**inicio**

**repita**

**escreva ("Entre com um valor inteiro positivo: ")**

**leia (A)**

**ate (A>0)**

**eh\_primo <- verdadeiro**

**para i de 1+1 ate A-1 faça**

**se (A%i=0) entao**

**eh\_primo <- falso**

**interrompa //causa uma saída imediata do laço**

**fimse**

**fimpara**

**se (eh\_primo e A<>1) entao**

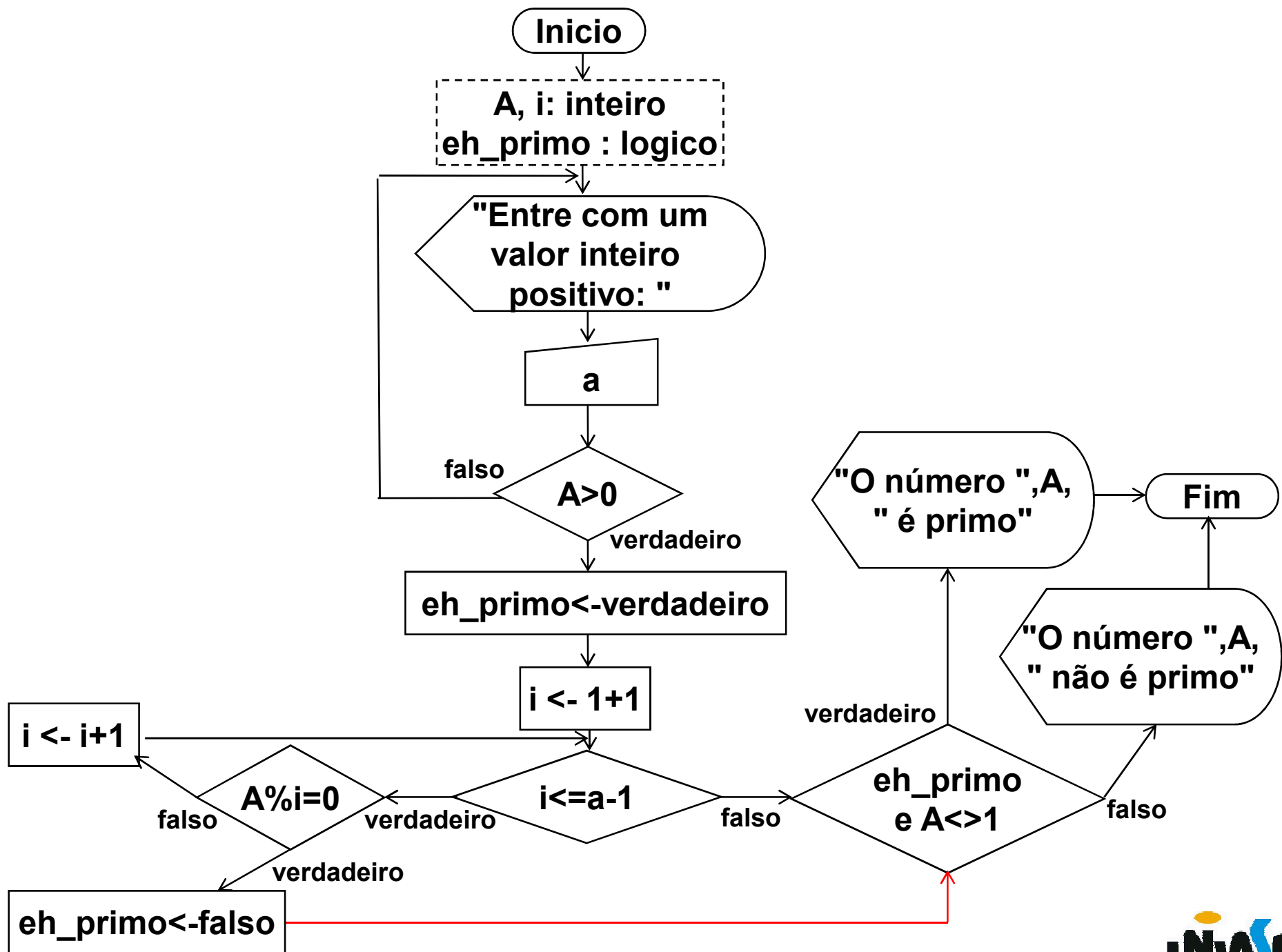
**escreva ("O número ",A," é primo")**

**senao**

**escreva ("O número ",A," não é primo")**

**fimse**

**fimalgoritmo**



## Estruturas de Controle de Fluxo

### 3. Laços de repetição (continuação)

Analisando melhor o problema de indicar se um determinado número  $n$  é ou não um número primo observamos o seguinte:

supondo que o  $n$  seja 9:

$$9/1 = 9$$

$$9/2 = 4.50$$

$$9/3 = 3 \quad \text{divisor} \leq \text{quociente}$$

$$9/4 = 2.25 \quad \text{divisor} > \text{quociente}$$

$$9/5 = 1.80$$

$$9/6 = 1.50$$

$$9/7 = 1.28$$

$$9/8 = 1.12$$

$$9/9 = 1$$

$$9^{0.5} = 3$$

**Não Primo**

supondo que o  $n$  seja 11:

$$11/1 = 11$$

$$11/2 = 5.50$$

$$11/3 = 3.67 \quad \text{divisor} \leq \text{quociente}$$

$$11/4 = 2.75 \quad \text{divisor} > \text{quociente}$$

$$11/5 = 2.20$$

$$11/6 = 1.83$$

$$11/7 = 1.57$$

$$11/8 = 1.37$$

$$11/9 = 1.22$$

$$11/10 = 1.1$$

$$11/11 = 1$$

$$11^{0.5} = 3.32$$

**Primo**

Testa apenas até

$$\left\lfloor \sqrt{n} \right\rfloor$$

## Estruturas de Controle de Fluxo

### 3. Laços de repetição (continuação)

#### Exercício 22:

Com base no que foi discutido no slide anterior construa um algoritmo que leia da entrada padrão um número inteiro positivo e retorne na saída padrão uma mensagem indicando se o número lido é ou não um número primo. As entradas devem ser validadas.