

## Estruturas de Controle de Fluxo

Os algoritmos desenvolvidos até o momento constituem uma sequência de ações que sempre são executadas em sua totalidade indiferente de qual(is) seja(m) o(s) valor(es) da(s) entrada(s).

Contudo, para a resolução de determinados problemas ou para a execução de determinadas tarefas é necessária a realização de um conjunto distinto de ações e este conjunto é definido com base em uma análise da(s) entrada(s).

Um exemplo cotidiano de uma destas situações é um algoritmo capaz de efetuar o cálculo do imposto de renda devido por um determinado contribuinte. Neste caso, dependendo da quantidade de dependentes, do valor de sua renda e outras fatores o cálculo será feito de formas distintas.

# Estruturas de Controle de Fluxo

Em função do que foi mencionado foram criadas as estruturas de controle de fluxo, as quais são fundamentais para a construção de algoritmos complexos. Estas permitem que o programador especifique a sequência de instruções que será executada.

## 1. Instrução condicional simples

Sintaxe: ...

se (*<expressão-lógica>*) entao  
*<seqüência-de-comandos>*

fimse

...

## **Estruturas de Controle de Fluxo**

**Pseudocódigo/Exercício** – Construa o pseudocódigo de um algoritmo para obter o resultado da divisão de dois números inteiros quaisquer.

**algoritmo “exercício 8”**

**var n1, n2: inteiro**

**res: real**

**inicio**

**escreva (“Digite o dividendo inteiro: ”)**

**leia (n1)**

**escreva (“Digite o divisor inteiro: ”)**

**leia (n2)**

**se (n2<>0) entao**

**res <- n1 / n2**

**escreva (“Resultado da divisão: ”, res)**

**fimse**

**fimalgoritmo**

## Estruturas de Controle de Fluxo

**Pseudocódigo/Exercício** – Construa o pseudocódigo de um algoritmo para obter o resultado da divisão de dois números inteiros quaisquer.

**algoritmo “exercício 8b”**

**var** n1, n2: inteiro

res: real

**inicio**

escreva (“Digite o dividendo inteiro: ”)

leia (n1)

escreva (“Digite o divisor inteiro: ”)

leia (n2)

**se** (n2<>0) **entao**

res <- n1 / n2

escreva (“Resultado da divisão: ”, res)

**fimse**

**se** (n2=0) **entao**

escreva (“Impossível dividir!”)

**fimse**

**fimalgoritmo**

# Estruturas de Controle de Fluxo

## 1. Instrução condicional composta

Sintaxe:

```
...  
se (<expressão-lógica>) entao  
    <seqüência-de-comandos-1>  
senao  
    <seqüência-de-comandos-2>  
fimse  
...
```

## **Estruturas de Controle de Fluxo**

**Pseudocódigo/Exercício** – Construa o pseudocódigo de um algoritmo para obter o resultado da divisão de dois números inteiros quaisquer.

**algoritmo “exercício 8c”**

**var n1, n2: inteiro**

**res: real**

**inicio**

**escreva (“Digite o dividendo inteiro: ”)**

**leia (n1)**

**escreva (“Digite o divisor inteiro: ”)**

**leia (n2)**

**se (n2=0) entao**

**escreva (“Impossível dividir!”)**

**senao**

**res <- n1 / n2**

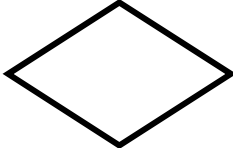
**escreva (“Resultado da divisão: ”, res)**

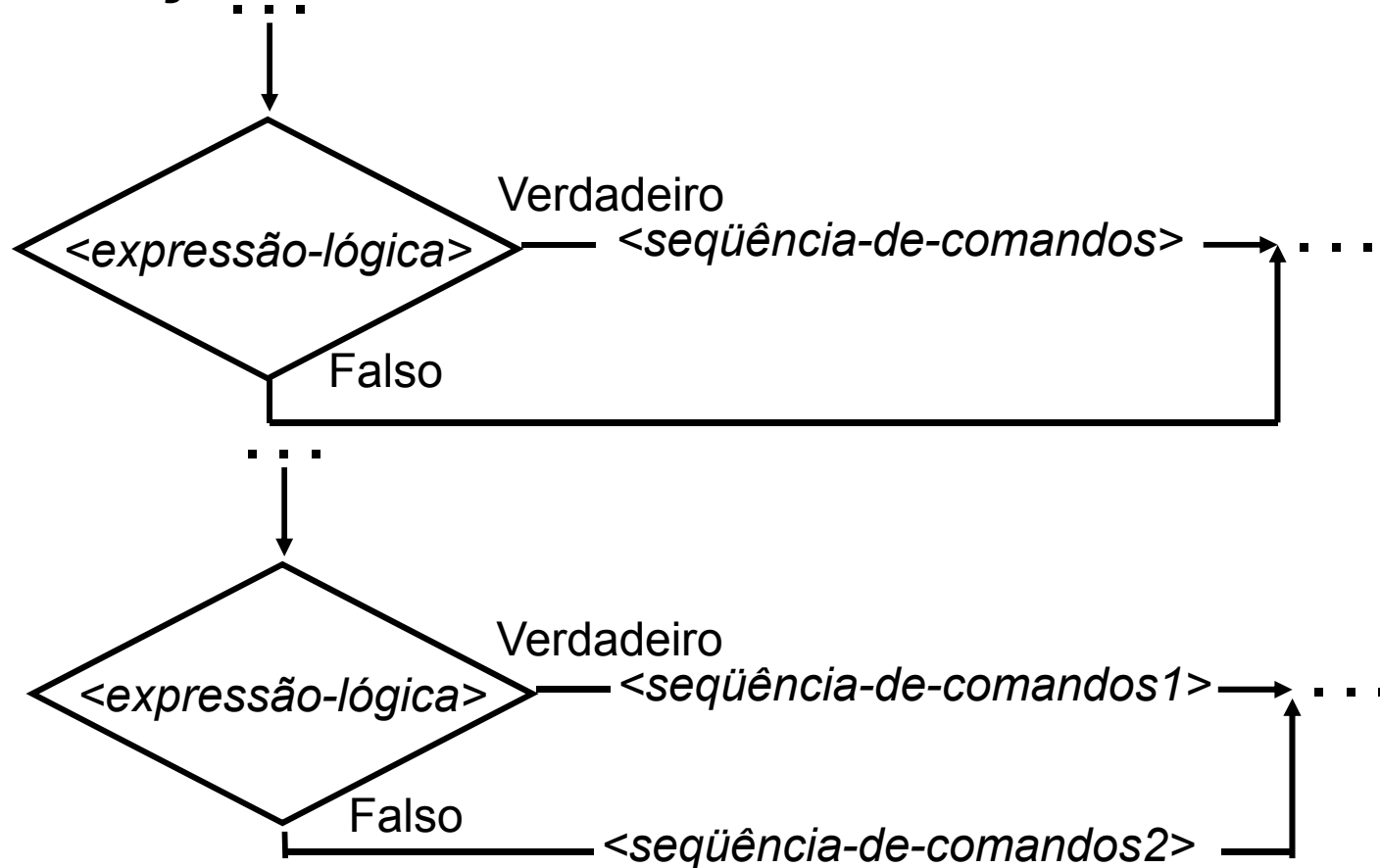
**fimse**

**fimalgoritmo**

# Estruturas de Controle de Fluxo

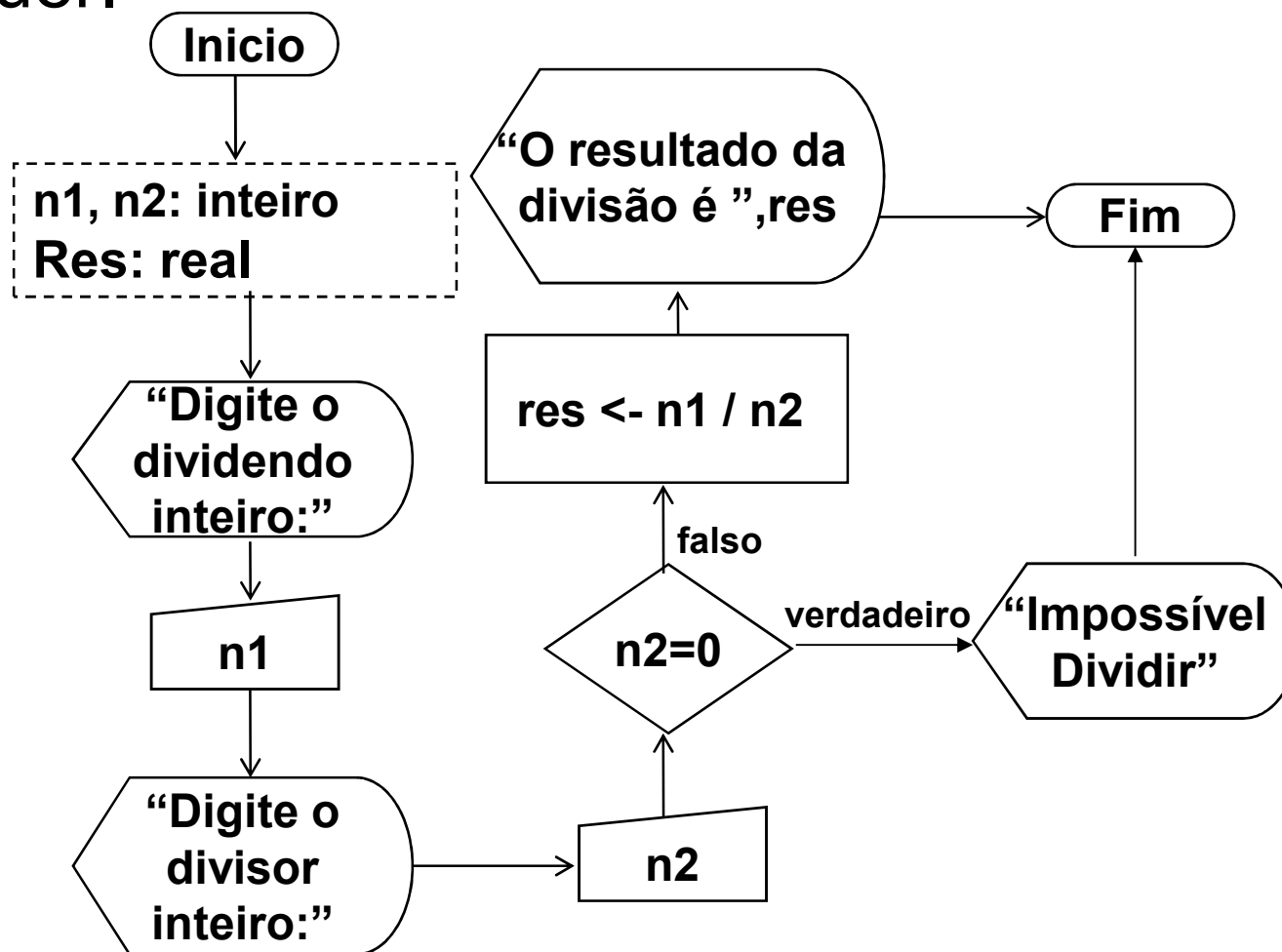
## Fluxograma

Vimos o símbolo  quando falamos sobre fluxograma. Este símbolo nos permite implementar a instrução “se entao” e “se entao senao”. Exemplo:



## Estruturas de Controle de Fluxo

**Fluxograma/Exercício** – Com base no que foi exposto construa um fluxograma para obter o resultado da divisão de dois números inteiros quaisquer.





## Estruturas de Controle de Fluxo

**Exercício 9** – Construa um algoritmo, representando-o com um pseudocódigo e com um fluxograma, que calcule o salário líquido de um funcionário. O algoritmo recebe através do teclado o salário bruto de um determinado funcionário, caso este seja inferior a R\$ 300,00 é descontado 5% em impostos, se o salário variar de R\$ 300,00 até 1.200,00 é descontado 10% em impostos, se esse for superior a R\$ 1.200,00 é descontado 15% em impostos. Ao final o algoritmo deve exibir o salário líquido do funcionário.

**algoritmo “exercício 9a”**

**var salario: real**

**inicio**

**escreva (“Entre com o salário bruto do funcionário: ”)**

**leia (salario)**

**se (salario<300.0) entao**

**salario <- salario\*0.95**

**fimse**

**se ((salario>=300) e (salario<=1200)) entao**

**salario <- salario\*0.90**

**fimse**

**se (salario>1200) entao**

**salario <- salario\*0.85**

**fimse**

**escreva (“O salário líquido do funcionário é: ”)**

**escreva (salario:6:2)**

**fimalgoritmo**

**algoritmo “exercício 9b”**

**var salario: real**

**inicio**

**escreva (“Entre com o salário bruto do funcionário: ”)**

**leia (salario)**

**se (salario<300.0) entao**

**salario <- salario\*0.95**

**senao**

**se ((salario>=300) e (salario<=1200)) entao**

**salario <- salario\*0.90**

**senao**

**salario <- salario\*0.85**

**fimse**

**fimse**

**escreva (“O salário líquido do funcionário é: ”)**

**escreva (salario:6:2)**

**fimalgoritmo**

