

# Algoritmos Computacionais:

Professor: Rodrigo Rocha  
Curso: Engenharia Civil  
Aula 6



# Agenda

- Estrutura Sequencial
- Estruturas de Seleção
  - Simples;
  - Composta;
  - Encadeada;
- Estruturas de Múltiplas Escolhas.

# Objetivos

- ▶ Entender a ideia de decisão e como implementá-la no computador;
- ▶ Compreender problemas com decisões múltiplas e sua implementação;
- ▶ Entender a representação de uma estrutura de decisão completa.

Próximo Assunto...

# ESTRUTURA SEQUENCIAL

# Estrutura Sequencial

- ▶ O Fluxo de Controle segue a mesma sequência linear da nossa escrita, ou seja:
  - De cima para baixo;
  - Da esquerda para direita
- ▶ Cada ação tem o intuito de:
  - Separar uma ação da outra
  - Indica que a próxima ação da sequência deve ser executada

# Estrutura Sequencial

## EXEMPLO DE UMA ESTRUTURA SEQUÊNCIAL

Algoritmo “Calcular a média”

**VAR**

// declaração de variáveis

N1, N2: **Real** // notas bimestrais

MA: **Real** // média anual

**inicio**

**Escreval**("Entre com as notas")

// entrada de dados

**leia** (N1)

**leia** (N2)

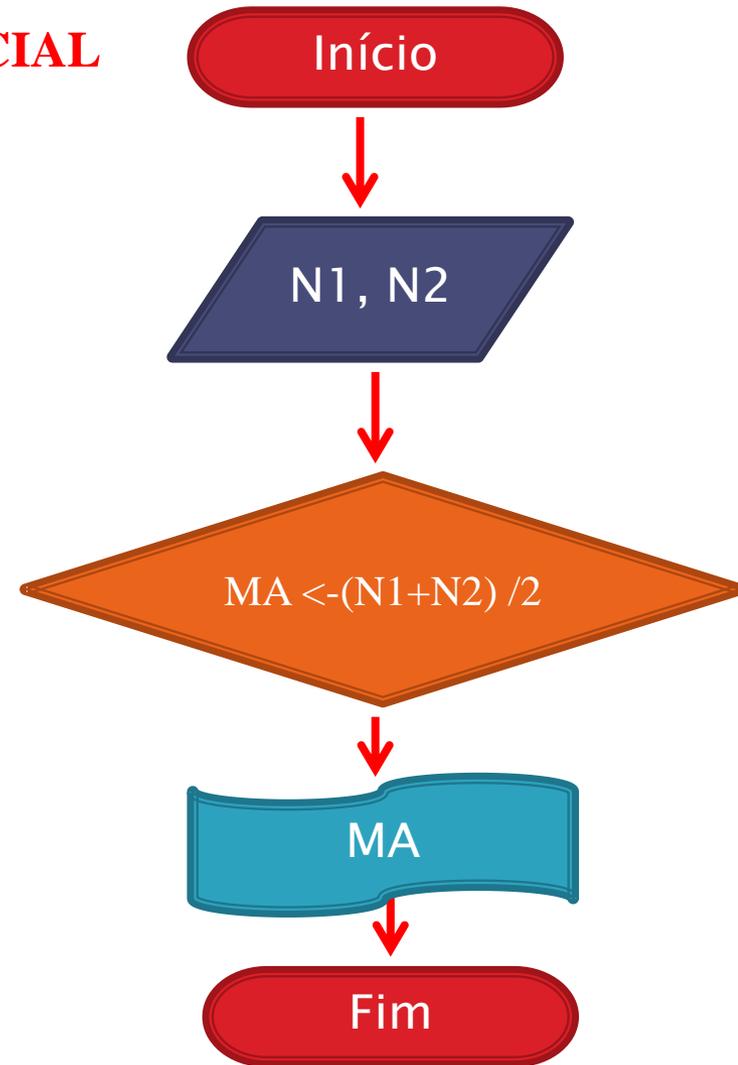
// processamento

MA <- (N1 + N2) / 2

// saída de dados

**escreva** (MA)

**fimalgoritmo**



**Próximo Assunto...**

# **ESTRUTURA DE SELEÇÃO SIMPLES**



# O que é Decisão para o Computador?

- ▶ Decidir: escolher se código / bloco será executado
- ▶ • Com base em quê?
  - Em uma proposição:
    - – verdadeiro → executa
    - – falso → não executa
- ▶ Exemplo:
  - – Imprimir “Reprovado”  
se  $N < 6.0$

# Estruturas de Seleção

- ▶ São aquelas que permitem alterar o Fluxo de Execução, de forma a selecionar qual parte deve ser executada
- ▶ Essa “decisão” de execução é tomada a partir de uma condição, que pode resultar apenas em Verdade ou Falsidade
- ▶ Uma condição é representada por expressões relacionais e/ou lógicas
- ▶ As estruturas de seleção podem ser classificadas em simples, compostas ou encadeadas

# Como Fica a Decisão no Código?

- Se  $\text{Nota} < 6.0$  imprime
- ▶ que aluno está reprovado.

- ▶ Código pseudocódigo:

- Algoritmo “nota”

Var

N: real

inicio

escreva(“Digite a nota: ”)

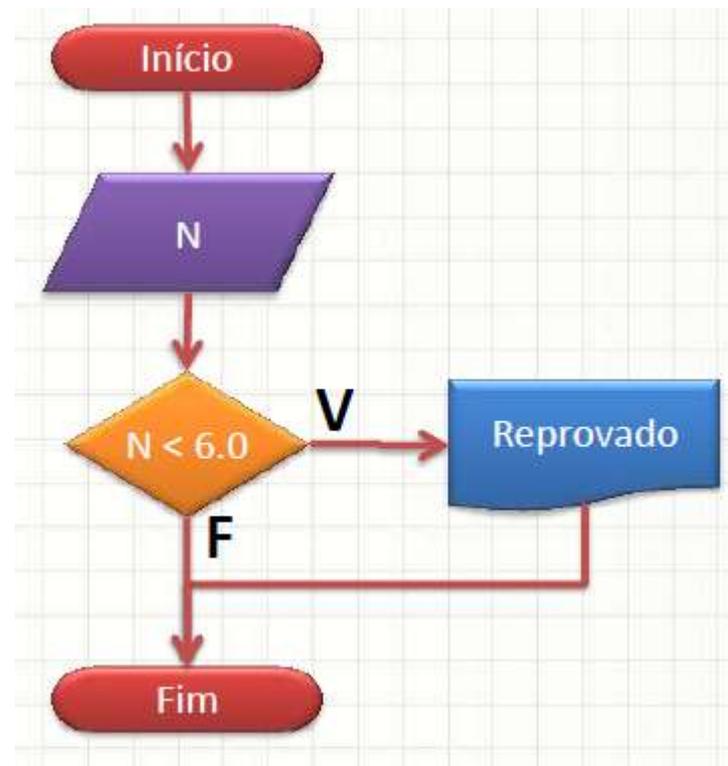
leia(N)

Se  $(N < 6)$  entao

    escreval(“Reprovado”)

fimse

fimalgoritmo



# Seleção Simples

```
▶ se <condição> entao  
    // início do bloco verdade  
    comando 1  
    comando 2  
    ...  
    comando n  
fimse; // fim do bloco verdade
```

```
se <proposição_lógica> entao  
    //código a executar para proposição é  
    verdadeira  
    comando 1  
    ...  
    comando n  
fimse
```

- ▶ Quando a <condição> / <proposição\_lógica> for **verdadeira** o “bloco verdade” é executado.
- ▶ Quando a <condição> / <proposição\_lógica> for **falsa** o “bloco verdade” **não** é executado.

# Seleção Simples

**Algoritmo** “Calcular a média”

**VAR**

// declaração de variáveis

N1, N2: **Real** // notas bimestrais

MA: **Real** // média anual

**inicio**

**Escreval**("Entre com as notas")

// entrada de dados

**leia** (N1)

**leia** (N2)

// processamento

MA <- (N1 + N2) / 2

// saída de dados

**escreva** (MA)

**se** (MA >= 7) **entao**

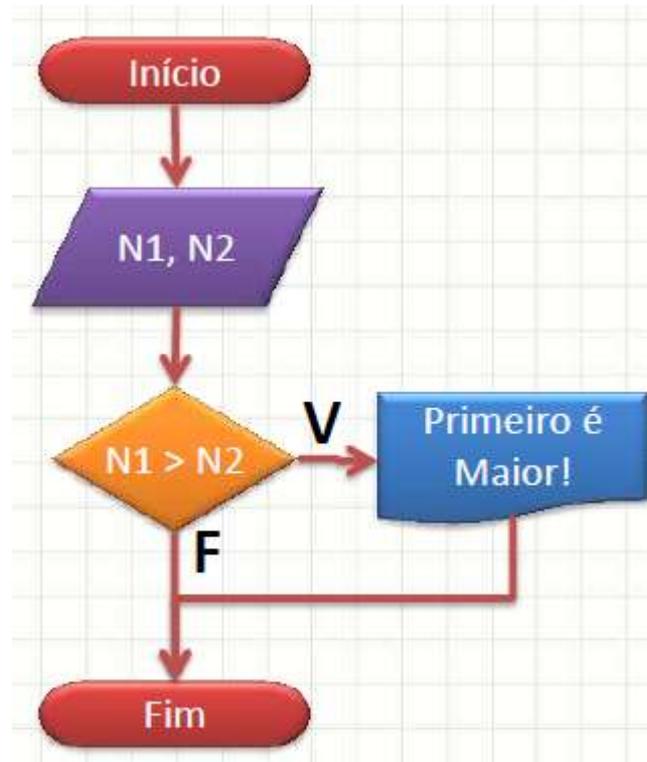
**escreval**("Aluno Aprovado !")

**fimse**

fimalgoritmo

# Vamos Testar...

- Faça um programa que lê dois números e responda se o primeiro é maior que o segundo número informado.



**Próximo Assunto...**

# **ESTRUTURA DE SELEÇÃO COMPOSTA**



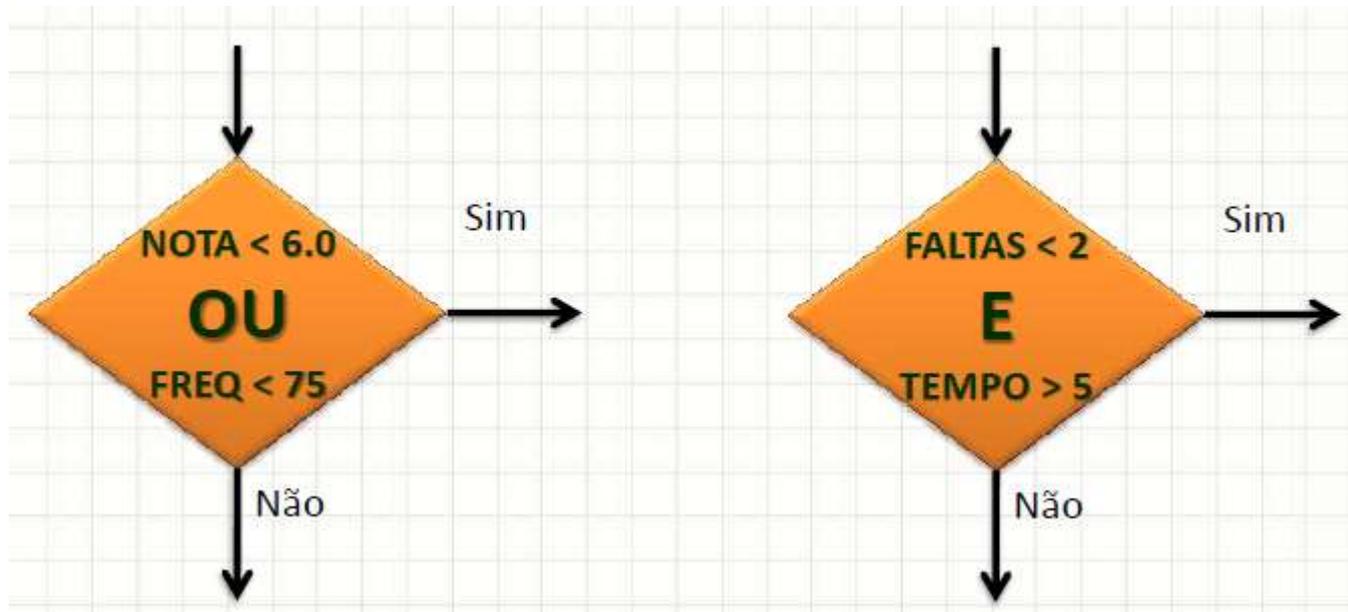
# Seleção Composta

```
Se <condição> entao //Condição verdade
    <ações (uma ou mais) a serem realizadas se a condição for verdadeira>
Senao //Condição falsidade (Caso contrário)
    <ações (uma ou mais) a serem realizadas se a condição for falsa>
fimse
```

- ▶ Quando a <condição> for **verdadeira** o “bloco verdade” é executado
- ▶ Quando a <condição> for **Falsa** o “bloco falsidade” (**caso contrário**) é executado.

# O que são decisões compostas?

- Decisão Simples: uma única comparação
- Decisão Composta: múltiplas regras
- Como?



# Seleção Composta

## EXEMPLO:

Algoritmo “Calcular a média”

VAR

// declaração de variáveis

N1, N2: Real // notas bimestrais

MA: Real // média anual

inicio

Escreval(“Entre com as notas“)

// entrada de dados

leia (N1)

leia (N2)

// processamento

se  $((N1 + N2) / 2) < 7$  entao

// saída de dados

escreval(“Reprovado!”)

senao

// saída de dados

escreval(“Aprovado!”)

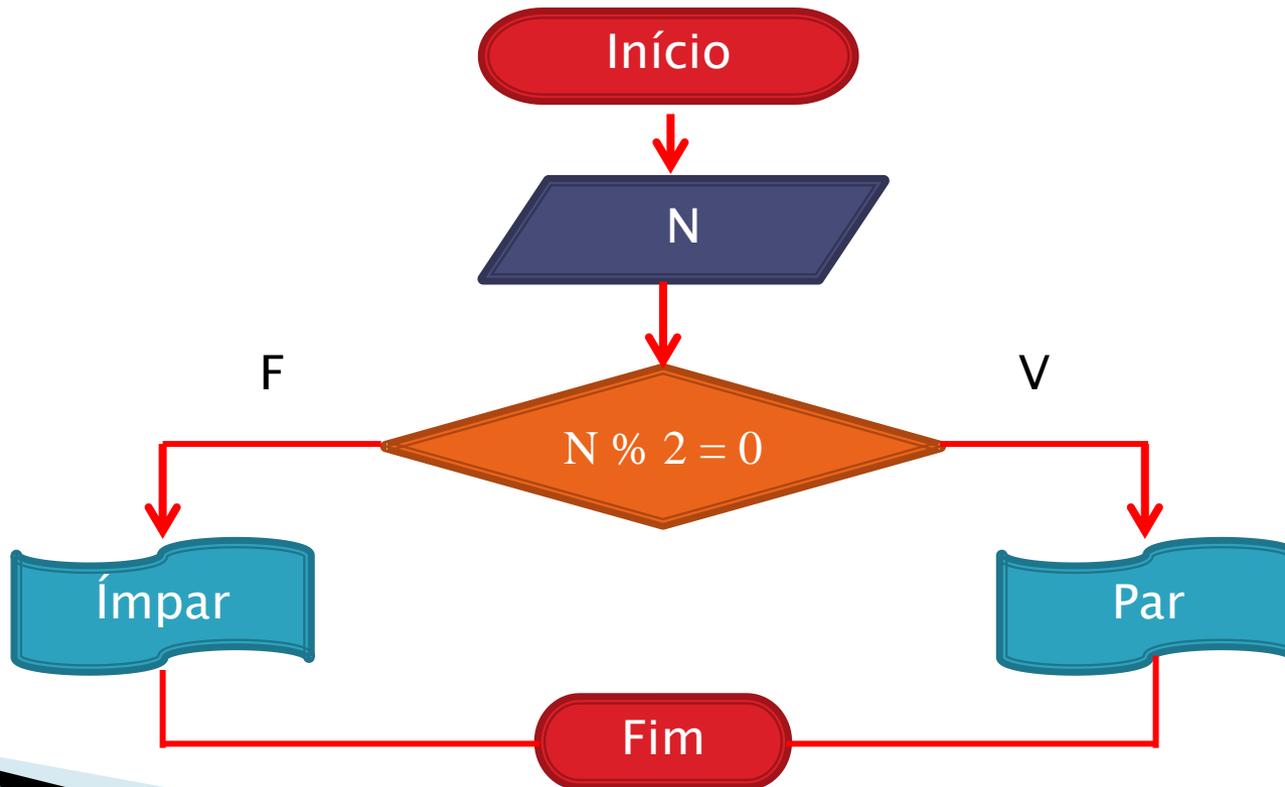
fimse

fimalgoritmo



# Vamos Testar...

- Faça um programa que lê um número e responda se o mesmo é par ou ímpar.



**Próximo Assunto...**

# **ESTRUTURA DE SELEÇÃO ENCADEADA**

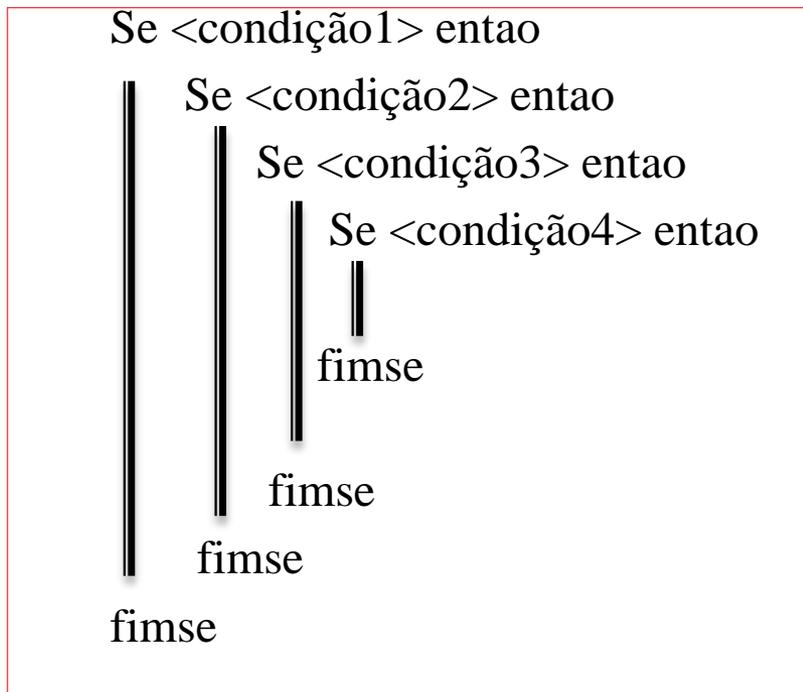


# Seleção Encadeada

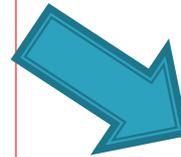
- ▶ Ocorre quando uma seleção tem como ação uma outra seleção
- ▶ Uma seleção encadeada pode ser elaborada das seguintes formas:
  - **Com “se-entao-se” aninhados**
    - Onde dentro de um “se-entao” (seleção simples) contém um ou mais “se-entao” (seleção simples), ou seja, uma seleção simples dentro da outra.
  - **Com se-senao-se aninhados**
    - Onde dentro de cada “se-senao” (seleção composta) contém um ou mais “se-senao” (seleção composta), ou seja, uma seleção composta dentro da outra.

# Seleção Encadeada com Seleção Simples

## ▶ EXEMPLO: “se-entao-se” aninhados



**OBS:** Os “se-entao” (seleção simples) mais internos só irão ser verificados se a proposição lógica do “se-então” imediatamente superior foi satisfeita (verdadeira).



Se (C1) e (C2) e (C3) e (C4) entao  
...  
fimse

# Seleção Encadeada com Seleção Composta

## ▶ EXEMPLO: se-senao-se aninhados

```
Se <condição1> entao
  ações..
Senao
  Se <condição2> entao
    ações..
  Senao
    Se <condição3> entao
      ações
    senao
      ações
    fimse
  fimse
fimse
```

**OBS:** As seleções compostas (“se-senao”) ” mais internas só irão ser verificadas se a proposição lógica imediatamente superior foi satisfeita (verdadeira).

**OBS2:** Dentro de uma seleção composta pode ocorrer uma ou mais seleções simples.

# Seleção Encadeada

## EXEMPLO PRÁTICO:

Algoritmo “Tipo de Triângulo”

VAR

A, B, C: inteiro // *tamanho dos lados*

Início

leia (A)

leia(B)

leia(C)

se  $(A < B + C)$  e  $(B < A + C)$  e  $(C < A + B)$  entao //verifica se os dados formam um triângulo

se  $((A = B)$  e  $(B = C))$  entao //Verifica se todos os lados são iguais

escreva (“Triangulo Equilátero”)

senao

se  $((A = B)$  ou  $(B = C)$  ou  $(A = C))$  entao //Verifica se dois lados são iguais

escreva (“Triângulo Isósceles”)

senao

escreva (“Triangulo Escaleno”)

fimse

fimse

senao

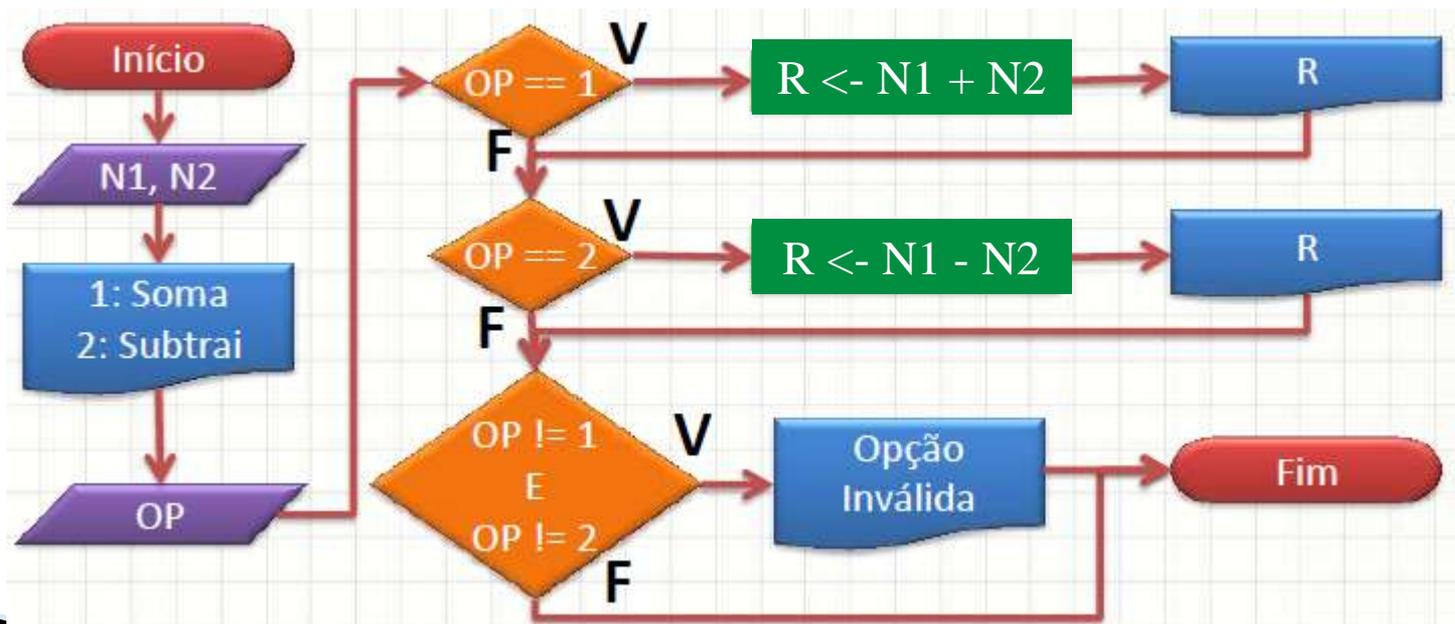
escreva (“Estes valores não formam um triângulo”)

fimse

fimalgoritmo

# Vamos Testar Novamente...

- ▶ Faça um programa que **lê dois números** e depois apresente um **menu com duas opções**: 1- Soma; 2- Subtrai e age de acordo com a opção escolhida. Se o usuário escolher uma **opção inválida**, deve apresentar **mensagem de erro**.



Próximo Assunto...

# ESTRUTURA DE SELEÇÃO DE MÚLTIPLAS ESCOLHAS



# Estrutura de Escolha

- E se um menu tiver 200 opções...
- Como ficaria o **se** final do exemplo anterior?
- ▶ Exemplo:
  - se (OP <>1 e OP<>2 e OP<>3 e ... e OP<>200) **entao**
  - ...
  - fimse**
- Inviável!
- Por isso existe uma estrutura **especial** para este tipo de menu!

# Seleção de Múltipla Escolhas

## ▶ Escolha...Caso

**escolha** < expressão-de-seleção >

**caso** < exp 1 > , < exp 2 > , ... , < exp n >

< lista-de-comandos-1 >

**caso** < exp 1 > , < exp 2 > , ... , < exp n >

< lista-de-comandos-2 >

**Outrocaso**

< lista-de-comandos-3 >

**fimescolha**

**OBS:** Como alternativa para o uso de seleção simples e composta para determinados casos, pode-se utilizar a seleção de múltiplas escolhas.

**OBS:** Os casos a serem avaliados devem pertencer ao mesmo domínio do tipo de dado escolhido como expressão-de-seleção.

# Seleção de Múltipla Escolha

**Algoritmo** "CLASSIFICAÇÃO DA REGIÃO DO PRODUTO"

**VAR**

Preco: **Real**

origem: **inteiro**

**Inicio**

**leia** (Preco)

**leia**(Origem)

**escolha** Origem

**caso** 1

**escreva** (Preco, “ – produto do Sul”)

**caso** 2

**escreva** (Preco, “ – produto do Norte”)

**caso** 3

**escreva** (Preco, “ – produto do Leste”)

**caso** 4,5,6

**escreva** (Preco, “ – produto do Oeste”)

**caso** 7, 8, 9

**escreva** (Preco, “ – produto do Sudeste”)

**outrocaso**

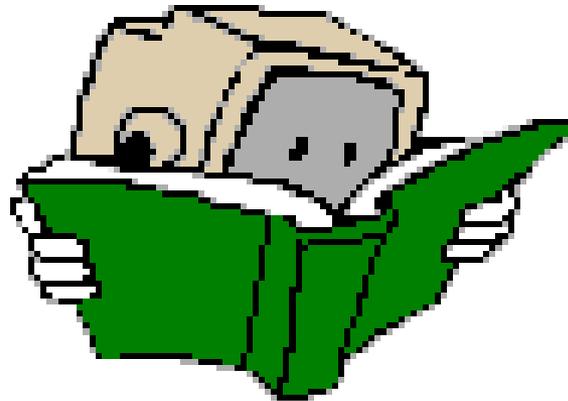
**escreva** (Preco, “ – produto importado”)

**fimescolha**

**fimalgoritmo**

# Perguntas?

???



# Referências

- ▶ Algumas das referências utilizadas:
  - TREMBLAY, J. P. **Ciência dos Computadores: Uma Abordagem Algorítmica**. Editora: McGraw-Hill, Cidade: São Paulo, 1983.
  - TONET, B., KOLIVER, C. **Introdução aos Algoritmos**. Universidade de Caxias do Sul – UCS – NAPRO – Núcleo de apoio aprendizagem de Programação.