



# Programação de Computadores I

## Estruturas de Decisão

PROFESSORA CINTIA CAETANO

# Introdução

---

- ▶ Na vida real tomamos decisões a todo o momento baseadas em uma situação existente.
- ▶ Em um algoritmo, chamamos esta situação de condição.
- ▶ Associada a uma condição, existirá uma alternativa possível de ações.
  
- ▶ Exemplo:
  - ▶ “Se chover hoje então levarei o guarda-chuvas.”

# Condição

---

- ▶ A condição nesta frase é “Se chover hoje”.
- ▶ Ela é uma expressão lógica, pois a pergunta é “Vai chover hoje?”
- ▶ Ela deve ser respondida com “Sim” ou “Não”.
- ▶ **Lembre-se:** em um algoritmo, toda condição tem que ser uma expressão lógica, algo que possa-se pensar como “VERDADEIRO” ou “FALSO”.

# Condição

---

- ▶ Se a condição for **verdadeira**, a ação a ser executada é “levarei o guarda-chuvas”, se a resposta à pergunta “Vai chover hoje?” for “Sim”.
- ▶ Então, em um algoritmo, as ações são um ou mais comandos que serão realizados apenas se a avaliação da condição resulta VERDADEIRO.

# Português Estruturado

---

- ▶ Vamos colocar agora a frase do exemplo anterior em outra forma, mais parecida com nosso Português Estruturado:

- ▶ **Exemplo:**

**Se “chover hoje” entao**

“levarei o guarda-chuvas”

**fimse**

# Se... Então...

---

- ▶ Tem por finalidade tomar uma decisão e efetuar um desvio no processamento, dependendo da condição atribuída ser verdadeira ou falsa.
- ▶ Sendo a condição verdadeira, será executada a instrução que estiver escrita após a instrução então.
- ▶ Se a instrução for falsa, serão executadas as instruções que estejam após as instruções consideradas verdadeiras.

# Se... Então...

---

## ▶ Sintaxe:

```
se <condição> entao
```

```
<ações (uma ou mais) a serem realizadas se a condição for verdadeira>
```

```
fimse
```

## ▶ Exemplo

```
Se (x > 10) entao
```

```
    writeln (“O valor da variável x é 10”);
```

```
fimse
```

# Exemplo 1

---

- ▶ Ler dois valores inteiros e independentemente da ordem em que foram inseridos, estes deverão ser exibidos em ordem crescente, ou seja, se forem fornecidos 5 e 3 respectivamente, deverão ser apresentados 3 e 5. O programa em questão deverá efetuar a troca dos valores entre as duas variáveis”

# Exemplo 1

---

## ▶ **Algoritmo**

1. Ler dois valores inteiros (estabelecer variáveis A e B);
2. Verificar se o valor de A é maior que o valor de B:
  - a. Se for verdadeiro, efetuar a troca de valores entre as variáveis;
  - b. Se for falso, pedir para executar o que está estabelecido no passo 3;
3. Apresentar os valores das duas variáveis.

# Exercício 1

---

- ▶ Faça um Algoritmo para calcular a área de um círculo, fornecido o valor do raio, que deve ser positivo.

```
1.  Algoritmo "Calcula Area do Circulo"
2.  VAR
3.  Area, Raio: Real
4.  inicio
5.  Escreval ("Entre com raio do círculo")
6.  Leia (Raio)
7.  Se Raio > 0 entao
8.      Area <- PI*(Raio^2)
9.      Escreva ("A área do círculo de raio ", Raio, " é ", Area)
10. fimse
11. Se Raio <= 0 entao
12.     Escreva ("Raio não pode ser nulo ou negativo!")
13. fimse
14. fimalgoritmo
```

# Se... Então...Senão...

---

- ▶ A instrução **se...entao...senao...** tem por finalidade tomar uma decisão e efetuar um desvio no processamento.
- ▶ Se a condição for **verdadeira** será executada a instrução logo abaixo do se.
- ▶ Sendo a condição **falsa**, será executada a instrução que estiver posicionada logo após a instrução senao.

# Se... Então...Senão...

---

## ► Sintaxe:

```
se <condição> entao
```

```
<ações (uma ou mais) a serem realizadas se a condição for verdadeira>
```

```
senao
```

```
<ações (uma ou mais) a serem realizadas se a condição for falsa>
```

```
fimse
```

## Exemplo 2

---

- ▶ Utilizando o se...entao...senao..., o algoritmo para calcular a área de um círculo, ficaria assim:

```
1.  Algoritmo "Calcula Area do Circulo"
2.  VAR
3.  Area, Raio: Real
4.  inicio
5.  Escreval ("Entre com raio do círculo")
6.  Leia (Raio)
7.  Se Raio > 0 entao
8.      Area <- PI*(Raio^2)
9.      Escreva ("A área do círculo de raio ", Raio, " é ", Area)
10. senao
11.     Escreva ("Raio não pode ser nulo ou negativo!")
12. fimse
13. fimalgoritmo
```

# Exemplo 2

---

```
1.  Algoritmo "Calcula Area do Circulo"  
2.  VAR  
3.  Area, Raio: Real  
4.  inicio  
5.  Escreval ("Entre com raio do círculo")  
6.  Leia (Raio)  
7.  Se Raio > 0 entao  
8.      Area <- PI*(Raio^2)  
9.      Escreva ("A área do círculo de raio ", Raio, " é ", Area)  
10. senao  
11.     Escreva ("Raio não pode ser nulo ou negativo!")  
12. fimse  
13. fimalgoritmo
```

## Exercício 2

---

- ▶ Ler dois valores numéricos e efetuar a adição. Caso o valor somado seja maior ou igual a 10, este deverá ser apresentado somando-se a ele mais 5. Caso o valor somado não seja maior ou igual a 10, esta deverá ser apresentado subtraindo-se 7.

# Exercício 2

---

## ▶ Algoritmo

1. Ler dois valores (variáveis A e B);
2. Efetuar a soma dos valores A e B, atribuindo o resultado da soma a uma variável X;
3. Verificar se X é maior ou igual a 10:
  - a. Se for verdadeiro, mostrar  $X+5$ ;
  - b. Se for falso, mostrar  $X-7$ .

# Operadores

---

- ▶ Podemos trabalhar com o relacionamento de duas ou mais condições dentro da instrução se...entao.
- ▶ Para estes casos, é necessário trabalhar com os operadores lógicos, vistos anteriormente.

# Operadores

---

```
1 algoritmo "teste_logico_e"  
2 Var  
3     num: inteiro  
4  
5 inicio  
6 Escreval ("Informe um número: ")  
7 Leia(num)  
8 Se ((num >= 0) e (num <= 100)) entao  
9     Escreva ("Número está na faixa de 10 e 100")  
10 Senao  
11     Escreva ("Número está fora da faixa de 10 e 100")  
12 fimse  
13 fimalgoritmo
```

# Operadores

---

```
1 algoritmo "teste_logico_ou"  
2 Var  
3     a, b: inteiro  
4  
5 inicio  
6 Escreval("Informe um número: ")  
7 Leia(a)  
8 Escreval("Informe outro número: ")  
9 Leia(b)  
10 Se ((a > 0) ou (b > 0)) entao  
11     Escreva ("Um dos números é positivo")  
12 fimse  
13 fimalgoritmo
```

# Operadores

---

```
1 algoritmo "teste_logico_not"
2 Var
3     a, b, c, x: inteiro
4
5 inicio
6 Escreva ("Informe o valor de A: ")
7 Leia(a)
8 Escreva ("Informe o valor de B: ")
9 Leia(b)
10 Escreva ("Informe o valor de X: ")
11 Leia(x)
12 Se nao (x > 5) entao
13     c <- (a+b)*x
14 senao
15     c <- (a-b)*x
16 fimse
17     Escreva ("O resultado é: ", c)
18 fimalgoritmo
```



# Escolha...Caso...

---

- ▶ Em algumas situações é necessário termos várias soluções ligadas a respostas diferentes.
- ▶ Temos então o comando de alternativa de múltipla escolha, o Escolha... Caso...
- ▶ Ele é utilizado para escolher apenas uma opção dentre várias alternativas.

# Escolha...Caso...

---

- ▶ O funcionamento deste comando obedece a seguinte regra:

```
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
```

# Exemplo

---

- ▶ Um determinado clube de futebol pretende classificar seus atletas em categorias e para isto ele contratou um programador para criar um programa que executasse esta tarefa.
- ▶ Para isso o clube criou uma tabela que continha a faixa etária do atleta e sua categoria. A tabela está demonstrada abaixo:
  - ▶ IDADE CATEGORIA
    - ▶ De 05 a 10 Infantil
    - ▶ De 11 a 15 Juvenil
    - ▶ De 16 a 20 Junior
    - ▶ De 21 a 25 Profissional

# Exemplo

---

```
1.  Algoritmo "CLASSIFICAÇÃO DE ATLETAS
2.  var
3.  nome, categoria  : caractere
4.  idade : inteiro
5.  inicio
6.  Escreva ("Nome do Atleta = ")
7.  Leia (nome)
8.  Escreva ("Idade do Atleta = ")
9.  Leia (idade)
10. Escolha idade
11. caso 5,6,7,8,9,10
12.     categoria <- "Infantil"
13. caso 11,12,13,14,15
14.     categoria <- "Juvenil"
15. caso 16,17,18,19,20
16.     categoria <- "Junior"
17. caso 21,22,23,24,25
18.     categoria <- "Profissional"
19. outrocaso
20.     categoria <- "INVALIDO"
21. Fimescolha
22. Escreva ("Categoria = ",categoria)
23. fimalgoritmo
```

