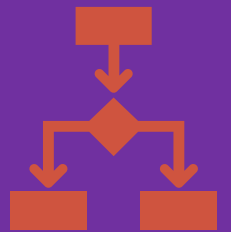


---

# Programação e Algoritmo

---

PSEUDOCÓDIGO



# Tipos de Dados

---

**Inteiro**  
**Real**  
**Caractere**  
**Lógico**

# Tipos de Dados

---

Tipo inteiro : dados numéricos positivos ou negativos, excluindo os fracionários

Tipo real : dados numéricos positivos ou negativos, incluindo os fracionários

Tipo caractere : Sequência contendo letras , números e símbolos especiais.  
Deve ser representadas entre aspas (“”)

Tipo Lógico: Dados com valores **verdadeiro** e **falso**.  
Só pode representar um destes valores.

# Tipos de dados

---

Exemplos de dados do tipo inteiro:

45                      0                      -35

Exemplos de dados do tipo real

45                      0                      -35                      0,45                      -2,54

Exemplos de dados do tipo caractere:

“João da Silva”      “Rua sabiá, 40”      “@#\$%”      “12340”

# VARIÁVEIS

---

Variáveis: Locais na memória do computador onde se guardam informações temporárias e que podem sofrer alterações no decorrer do programa.

# VARIÁVEIS

---

Regras para dar nome a uma variável:

1. Podem conter, mas não podem começar por números
2. Não devem conter símbolos especiais (exceção para o símbolo \_)
3. Não poderá conter brancos
4. Não poderá ter como nome uma palavra reservada da linguagem de programação usada

# Declaração da variável

---

Var

Variavel1 : *tipo*

Variavel2 : *tipo*

*Tipo* : Tipo de dado que a variável deverá conter(inteiro, real, caractere ou lógico)

# Operadores - Atribuição

---

Operador	Significado	Exemplo
$\leftarrow$	atribuição	$A \leftarrow b$

Colocando valor na variável **a**:

$a \leftarrow 5$

(pode-se ler assim: **a** que recebe cinco)

Colocando o valor de **a** em **b**:

$b \leftarrow a$



# Operadores - Atribuição

---

Troca de valores entre **a** e **b**:

$x \leftarrow b$

$b \leftarrow a$

$a \leftarrow x$

# Operadores Aritméticos

---

Operadores	Significado	Exemplo
+	Adição	$a + b$
-	Subtração	$a - b$
*	Multiplicação	$a * b$
/	Divisão fracionária	$a / b$
\	Divisão inteira	$a \setminus b$
%	Resto da divisão	$a \% b$

# Operadores Aritméticos

---

Exemplos:

Somando 1 em **a**:  $a \leftarrow a + 1$

Somando **a** e **b** resultando **s**:  $s \leftarrow a + b$

Subtraindo 1 de **b**:  $b \leftarrow b - 1$

Subtraindo **b** de **a** resultando **d**:  $d \leftarrow a - b$

Multiplicando **a** por 5:  $a \leftarrow a * 5$

Dividindo **b** por 2,5 resultando **q**:  $q \leftarrow b / 2.5$

Obtendo o resto em **r** da divisão de **b** por 11:  $r \leftarrow b \% 11$

# Operadores relacionais

---

Operadores	Significado	Exemplo
>	Maior que	$a > b$
<	Menor que	$a < b$
=	Igual a	$a = b$
<>	diferente	$a <> b$
>=	Maior ou igual a	$a >= b$
<=	Menor ou igual a	$a <= b$

# Operadores lógicos

---

Operadores lógicos
OU
E
NÃO

# Operadores lógicos - OU

---

Condição 1	Condição 2	Resultado
Verdadeiro	Verdadeiro	Verdadeiro
Verdadeiro	Falso	Verdadeiro
Falso	Verdadeiro	Verdadeiro
Falso	Falso	Falso

# Operadores lógicos - E

---

Condição 1	Condição 2	Resultado
Verdadeiro	Verdadeiro	Verdadeiro
Verdadeiro	Falso	Falso
Falso	Verdadeiro	Falso
Falso	Falso	Falso

# Operadores lógicos - NÃO

---

Condição	Resultado
Verdadeiro	Falso
Falso	Verdadeiro



# Programa – Forma Geral

---

## Codificação de um algoritmo

algoritmo *<nome do programa>*

Var

*<declaração de variáveis>*

Inicio

*<comandos>*

Fim

# Entrada e saída de dados

---

O que é	Comando	Exemplo
Entrada de dados	Leia( <i>variável</i> )	Leia(a)
Saída de dados	Escreva( <i>variável</i> ) Escreva( <i>texto</i> ) Escreva( <i>texto, variável</i> )	Escreva(a) Escreva("Olá Mundo") Escreva("Valor de a:", a)

# Entrada e saída de dados

---

algoritmo "LeEscreve"

Var

nome : caractere

idade : inteiro

nota : real

Início

Leia(nome)

Leia(idade)

Leia(nota)

Escreva( "Nome de aluno: ", nome)

Escreva("Idade: ", idade)

Escreva("Nota da prova", nota)

Fim

# Estruturas de decisão

## se... então... senão

---

### **Seleção simples**

*Se condição* então

*Comando(s)*

FimSe

# Estruturas de decisão

## se... então... senão

---

- algoritmo "ImpXMaior"
- Var
- $x$  : inteiro
- $y$  : inteiro
- Início
- Leia( $x$ )
- Leia( $y$ )
- Se  $x > y$  então
- Escreva( $x$ )
- FimSe
- Fim

Este programa imprime o valor de  $x$ , se este for maior que  $y$

# Estruturas de decisão

## se... então... senão

---

### **Seleção Composta**

*Se condição então*  
*comando(s)*

*Senão*  
*comando(s)*

FimSe

# Estruturas de decisão

## se... então... senão

---

algoritmo "ImpMaior"

Var

  x : inteiro

  y : inteiro

Início

  Leia(x)

  Leia(y)

  Se  $x > y$  então

    Escreva(x)

  Senão

    Escreva(y)

  FimSe

Fim

Este programa imprime o maior valor ( **x** ou **y** )

# Estruturas de decisão

## caso ... seja

---

### Seleção Múltipla

Escolha *opção*

caso *opção(ões)*  
    *Comando(s)*

caso *opção(ões)*  
    *Comando(s)*

...

outro caso  
    *Comando(s)*

fimescolha



# Estruturas de decisão

## caso ... seja

---

```
algoritmo "Times"
var time: caractere
inicio
    escreva ("Entre com o nome de um time de futebol: ")
    leia (time)
    escolha time
        caso "Flamengo", "Fluminense", "Vasco", "Botafogo"
            escreva ("É um time carioca.")
        caso "São Paulo", "Palmeiras", "Santos", "Corinthians"
            escreva ("É um time paulista.")
        outrocaso
            escreval ("É de outro estado.")
fimescolha
finalgoritmo
```

# Estruturas de repetição

---

<b>Enquanto ... faça</b>
Enquanto <i>condição</i> faça <i>Comando(s)</i> FimEnquanto

# Estruturas de repetição

## Enquanto Faça

---

```
Exemplo
algoritmo "multiplos"
Var
    num, produto : inteiro
    cont : inteiro

Inicio
    cont ← 0
    Enquanto cont < 5 faça
        leia(num)
        produto ← num * 3
        escreva(produto)
        cont ← cont + 1
    FimEnquanto
Fim
```

# Estruturas de repetição

---

<b>Repita até</b>
Repita <i>Comando(s)</i> Até Condição

# Estruturas de repetição

## Repita até

---

```
Exemplo
algoritmo "multiplos"
Var
    num, produto : inteiro
    cont : inteiro

Inicio
    cont ← 0
    Repita
        leia(num)
        produto ← num * 3
        escreva(produto)
        cont ← cont + 1
    Até cont = 5
Fim
```

# Estruturas de repetição

## Para Faça

Para *controle* de *valor\_inicial* até *valor\_final* passo *valor* faça  
    *Comando(s)*  
FimPara

# Estruturas de repetição

## Para faça

---

Exemplo

algoritmo "multiplos"

Var

num, produto : inteiro

cont : inteiro

Inicio

Para cont de 1 até 5 passo 1 faça

leia(num)

produto  $\leftarrow$  num \* 3

escreva(produto)

FimPara

Fim