

## GUIA DE REFERÊNCIA – ALGORITMOS EM PORTUGUES ESTRUTURADO

PROFA. JANE MARIA DOS SANTOS EBERSON

ESTRUTURA BÁSICA PARA CONSTRUÇÃO DOS ALGORITMOS EM PORTUGUES ESTRUTURADO.  
FONTE: ANA FERNANDA GOMES ASCENCIO DE CAMPOS E EDILENE APARECIDA VERENUCHI.  
FUNDAMENTOS DE PROGRAMAÇÃO DE COMPUTADORES: ALGORITMOS, PASCAL, C/C++  
(PADRÃO ANSI) E JAVA.

### Tipo de dados básicos

- Numérico: Entram os dados do conjunto de números Inteiro e Reais
- Lógicos: Podem assumir valores verdadeiro ou falso
- Caracteres (Literais): Dados formados por um único caractere ou por uma cadeia de caracteres.

### Declaração de variáveis

```
DECLARE Num NUMÉRICO
        Texto CARACTERE
        Compare LÓGICO
```

### Atribuição de valores em variáveis

```
Num <- 4
Texto <- "aula"
Compare <- verdadeiro
```

### Comando de entrada

```
LEIA Num
```

### Comando de saída

```
ESCREVA Texto
```

### Operadores Aritméticos

OPERADOR	FUNÇÃO
+	Soma
-	Subtração
/	Divisão
*	Multiplicação
** OU ^	Exponenciação
MOD	Resto da divisão inteira
DIV	Quociente da divisão inteira

### Operadores Relacionais

OPERADOR	FUNÇÃO
=	Igual
<>	Diferente
>=	Maior ou igual que
<=	Menor que ou igual

### Operadores Lógicos

OPERADOR	FUNÇÃO
E	Conjunção
OU	Disjunção (não exclusiva)
XOU	Disjunção (exclusiva)
NÃO	Negação

### Estrutura Condicional

#### Simple

```
SE condição
ENTÃO comando
```

#### Composta

##### Tipo 1

```
SE condição
ENTÃO comando1
SENÃO comando2
```

##### Tipo 2

```
SE condição
ENTÃO INÍCIO
        Comando1
        Comando2
FIM
SENÃO INÍCIO
        Comando3
        Comando4
FIM
```

### Estrutura de Repetição

#### Laço PARA

```
PARA i <-valor_inicial ATÉ valor_final
FAÇA PASSO n
INÍCIO
        Comando1
        Comando2
.
.
.
FIM
```

#### Estrutura ENQUANTO

```
ENQUANTO condição FAÇA
Comando1
```

#### Ou

```
ENQUANTO condição FAÇA
INÍCIO
        Comando1
        Comando2
FIM
```

#### Estrutura REPITA

```
REPITA
        Comandos
ATÉ condição
```

## Estrutura de dados

### Vetor

```
DECLARE nome[tamanho] tipo
```

### Matriz

```
DECLARE nome[linha,coluna] tipo
```

## Sub-Rotinas

### Chamada da sub-rotina do tipo função:

```
xpto <- rotina(val)
```

### Criação da subrotina do tipo função:

**Sub-rotina** rotina(xpto tipoVariavel)

Conjunto de instruções

**RETORNE** var

Fim da sub-rotina

### Chamada da sub-rotina do tipo

**procedimento:**

```
rotina(val)
```

### Criação da subrotina do tipo

**procedimento:**

**Sub-rotina** rotina(xpto tipoVariavel)

Conjunto de instruções

Fim da sub-rotina

## Estrutura de dados heterogêneas

### Registro

#### Declaração

```
DECLARE nome_da_variável_registro  
REGISTRO (nome_campo1 TIPO_DO_CAMPO1,  
nome_campo2 TIPO_DO_CAMPO2, ...,  
nome_campoN TIPO_DO_CAMPOn)
```

## Manipulação do registro

```
nome-do-registro.campo
```

## Arquivo

### Declaração

```
Nome-do-arquivo = ARQUIVO SEQUENCIAL DE  
nome-do-registro
```

### Manipulação

```
ABRIR-ARQUIVO (nome-do-arquivo)
```

```
FECHAR-ARQUIVO (nome-do-arquivo)
```

## Incluir novo registro no arquivo

```
INCLUIR nome-do-registro
```

## Excluir registro no arquivo

```
EXCLUIR nome-do-registro
```

## Ler um registro no arquivo

```
LEIA nome-do-registro
```