

Guia de Referência (resumo) para Linguagem de Programação do VISUALG

(Prof. Bruno B. Boniati – Colégio Agrícola de Frederico Westphalen – Universidade Federal de Santa Maria)

Estrutura básica de uma aplicação

```
algoritmo "teste" // Estrutura básica de um programa
```

```
funcao funcao_exemplo : real
inicio
    retorne 10
fimfuncao
```

```
procedimento procedimento_exemplo(f : real)
inicio
    f <- 10;
fimprocedimento
```

```
var a : inteiro
```

```
inicio
    //corpo principal do programa
finalgoritmo
```

Tipos Primitivos

Tipo	Propósito
inteiro	números sem casas decimais
real	números com casas decimais
caractere	sequências de caracteres (palavras, frases)
logico	VERDADEIRO ou FALSO

Operadores

Tipo	Operador	Propósito	Exemplo
Aritméticos	+	Adição para números e concatenação para caracteres.	a := 4 + 1 // 5
	-	Subtração	a := 4 - 1 // 3
	*	Multiplicação	a := 2 * 4 // 8
	/	Divisão	a := 8 / 2 // 4
	%	Módulo (resto da divisão)	a := 5 MOD 2; // 1
	\	Quociente da divisão inteira	a := 9 \ 2 //Resultado 4
	^	Potenciação	a := 5 ^ 2 //Resultado 25 5 ²
Atribuição	<- :=	Atribuição simples	a <- 50 a := 50
Lógicos	e	“e” lógico	(a > 1) e (b < 1)
	ou	“ou” lógico	(a > 1) ou (b < 1)
	nao	não (inversão)	nao (a > 2)
Relacionais (Comparação)	=	igual a	(a = 0)
	<>	diferente de	(a <> 0)
	<	menor que	(a < 0)
	>	maior que	(a > 0)
	<=	menor ou igual a	(a <= 0)
	>=	maior ou igual a	(a >= 0)

Entrada e Saída

escreva(lista de constantes/variáveis/expressões)

Comando para saída de valores (escreve na tela as expressões passadas entre parêntesis)

Ex.: escreva("O valor de A é ",a, " e o valor de B é ", b)

escreval(lista de constantes/variáveis/expressões)

Função para saída de valores que adiciona uma quebra de linha ao final do resultado.

Ex.: escreva("O valor de A é ",a, " e o valor de B é ", b)

leia(lista de variáveis)

Função para capturar e armazenar valores fornecidos via teclado.

Ex.: leia(a);
leia(a,b);

Comandos da Linguagem		
Comando	Propósito	Sintaxe
Declaração de variável	Declaração de variável	var nome_variavel : tipo
Bloco	Marcar um bloco de cód.	inicio fim??? (cada bloco tem um fim específico: fimalgoritmo, fimfuncao, fimprocedimento, fimse, fimescolha, fimpara, fimenquanto)
se	Comando condicional simples	se (a > b) então escreval("A é maior que B") senao escreval("A é igual ou menor que B") fimse
escolha	Comando condicional múltiplo	escolha i caso 0 escreval("ZERO") caso 1 escreval("UM") caso 2 escreval("DOIS") outrocaso escreval("OUTRO NUMERO") fimescolha
enquanto	Laço de repetição com pré validação	enquanto (i <= 10) faca escreval(i) i := i+1 fimenquanto
repita	Laço de repetição com pós validação	repita escreval(i) i := i+1 ate (i >= 10)
para	Laço de repetição simplificado	para i de 1 ate 10 faca escreval(i) fimpara
interrompa	Interrompe o laço de repetição	interrompa
Sub-rotinas	Funções	funcao area(altura, base : real) : real inicio retorne altura * base fimfuncao
	Procedimentos	procedimento area(altura, base : real) inicio escreval("A área é:", altura * base); fimprocedimento
Vetores	Variáveis unidimensionais	v: vetor [1..10] de real //Vetor de inteiros //v[1] é o primeiro elemento e v[10] o último v[1] := 55;
Matrizes	Variáveis multidimensionais	matriz: vetor [1..4,1..3] de inteiro //Tabela com 4 linhas e 3 colunas matriz[2,2] :=

Funções utilitárias

- **aleatorio** <valor1>, <valor2> - ativa a geração de valores aleatórios que substituem a digitação de dados entre os valores 1 e 2. O comando **aleatorio off** deliga a geração de valores.
- **arquivo** <nome_arquivo.txt> - direciona a origem da entrada dos comandos leia para uma arquivo.
- **timer** <tempo_em_milissegundos> - altera o fluxo de execução dos comandos com o efeito de "camera lenta". O comando **timmer off** desliga este efeito.
- **pausa** - interrompe incondicionalmente o fluxo de execução.
- **debug** <expressão lógica> - interrompe o fluxo de execução se a expressão lógica for verdadeira.
- **eco on | off** - liga ou desliga a impressão dos dados de entrada na saída-padrão.
- **cronometro on | off** - liga ou desliga a contagem de tempo da execução do algoritmo.