

ESTUDO DIRIGITO DE ALGORITMO E LOGICA EM PROGRAMAÇÃO (AULA 1-3)

Tipo de variável (caractere, inteiro, real, logico(v/f),);

Funções básicas (nome <- "Silva", "escreva(nome), leia(x), operadores lógicos (<>, >=, <=, =))

Estruturas Condicionais (se, então; se, senão; escolha - caso)

Estruturas de Repetição (enquanto; repita; para)

Funções básicas

Leia (): usado para descrever o valor da variável, seja ela caractere, inteiro ou real

Exemplo: "leia (mes)"

Escreva ("...", ...): usado como pretexto para que o usuário escreva de acordo com a situação, quando se coloca aspas dentro dos parênteses, ela se torna uma frase explicativo, fora dos parênteses, o programa lê a função variável.

Exemplo: escreva ("digite o mes ", maio)

Escreval: tem a mesma função do escreva, mas pula linha.

Atribuição (<-): usado para indicar que a variável é resultante de uma outra var.

Exemplo: peso <- 30

Operadores Relacionais

=, <, >, <=, >=, <>	Respectivamente: igual, menor que, maior que, menor ou igual a, maior ou igual a, diferente de. São utilizados em expressões lógicas para se testar a relação entre dois valores do mesmo tipo. Exemplos: 3 = 3 (3 é igual a 3?) resulta em VERDADEIRO ; "A" > "B" ("A" está depois de "B" na ordem alfabética?) resulta em FALSO.
---------------------	--

Importante: No VisuAlg, as comparações entre *strings* **não diferenciam** as letras maiúsculas das minúsculas. Assim, "ABC" é igual a "abc". Valores lógicos obedecem à seguinte ordem: FALSO < VERDADEIRO.

Operadores Lógicos

nao	Operador unário de negação. nao VERDADEIRO = FALSO, e nao FALSO = VERDADEIRO. Tem a maior precedência entre os operadores lógicos. Equivale ao NOT do Pascal.
ou	Operador que resulta VERDADEIRO quando um dos seus operandos lógicos for verdadeiro. Equivale ao OR do Pascal.
e	Operador que resulta VERDADEIRO somente se seus dois operandos lógicos forem verdadeiros. Equivale ao AND do Pascal.
xou	Operador que resulta VERDADEIRO se seus dois operandos lógicos forem diferentes, e FALSO se forem iguais. Equivale ao XOR do Pascal.

ESTRUTURAS CONDICIONAIS:

Se (...) ENTAO- insere o comando 'se' sem a alternativa 'senao', ela dá uma instrução do "operador" caso o resultado seja correto ao que foi dito

Estrutura:

[se* então

Fimse]

Se(...)senao- insere o comando 'se' completo, sendo o 'senao' opcional, ela dá uma instrução do "operador". Realiza 'Então' caso o resultado seja correto ao que foi dito; realiza 'Senão' caso o resultado seja oposto

Estrutura:

[Se*então

Senão

Fimse]

Exemplo:

algoritmo "SacarDinheiro"

var

SaldoDisponivel : REAL

ValorDoSaque : REAL

inicio

SaldoDisponivel := 1000 //Assumimos que há 1000 reais de saldo na conta disponível para saque

ESCREVA ("Informe o valor do Saque: ")

LEIA (ValorDoSaque)

SE ValorDoSaque <= SaldoDisponivel ENTAO

SaldoDisponivel := SaldoDisponivel - ValorDoSaque

ESCREVAL ("Sacando R\$ ", ValorDoSaque, ".")

SENAO

ESCREVAL ("O valor solicitado é maior que o valor disponível para saque!")

FIMSE

ESCREVAL ("Saldo disponível: R\$ ", SaldoDisponivel)

fimalgoritmo

Escolha*Caso- insere o comando 'escolha' (com a clausura 'outrocaso'), quando outras verificações tenham falhado, permitindo ir direto ao bloco de notas dos códigos.

Estrutura:

[Escolha*Caso

Outrocaso

Fimescolha]

Exemplo:

escolha (operador)

caso "+" // Caso o valor do operador for +

resultado <- numero1 + numero2 // O resultado recebe a soma dos números

caso "-" // Caso o valor do operador for -

resultado <- numero1 - numero2 // O resultado recebe a subtração do

numero1 pelo numero2

caso "*" // Caso o valor do operador for *

resultado <- numero1 * numero2 // O resultado recebe a multiplicação dos

números

caso "/" // Caso o valor do operador for /

resultado <- numero1 / numero2 // O resultado recebe a divisão do numero1

pelo numero2

fimescolha

ESTRUTURAS DE REPETIÇÃO:

Para...faça: essa estrutura repete uma sequência de comandos um número de vezes.

Estrutura:

para <variável> de <valor-inicial> ate <valor-limite> [passo <incremento>] faça

 <sequência-de-comandos>

fimpara

Exemplo: tabuada

```
PARA contador DE 1 ATE 10 FACA
    multiplicacao <- numero * contador
    escreval(numero, " X ", contador, " = ", multiplicacao)
FIMPARA
```

Enquanto...Faça: esta estrutura repete uma sequencia de comandos enquanto uma determinada condição (especificada através de uma expressão logica) for satisfeita.

Estrutura:

enquanto <expressão-lógica> faça

 <sequencia-de-comandos>

Fimenquanto

Exemplo:

```
ENQUANTO (eh_primo = VERDADEIRO) FACA
    escreva(numero)

    se ( primo % numero) = 0 ) entao
        eh_primo <- FALSO
        escreval(" Não é primo")
    fimse

    numero<- numero+ 1
FIMENQUANTO
```

Repita...até: esta estrutura repete uma sequencia de comandos até que uma determinada condição (especificada através de uma expressão logica) seja satisfeita.

Estrutura:

Repita

 <seqüência-de-comandos>

ate <expressão-lógica>

Exemplo:

REPITA

```
    escreva("Digite um número para ser somado: ")
    leia(numero)
    soma <- soma + numero
    escreva("Deseja continuar digitando? [S/N]")
    leia(resposta)
    ATE (resposta <> "S")
```

repita

```
    incremento<-incremento-1

    soma<-soma+incremento

    escreval(incremento, " -- ", soma)

    ate(incremento = 1)
```