

Curso Completo de Algoritmos e Lógica de Programação

Capítulo: Estrutura sequencial

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

Visão geral sobre o capítulo

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

Conteúdo

- Expressões aritméticas
- Variáveis e tipos básicos
- Três operações básicas de programação
 - Entrada de dados
 - Processamento de dados
 - Saída de dados
- Funções matemáticas

Por que o nome "estrutura sequencial"?

Para enfatizar que os comandos do algoritmo **executam em sequência, de cima para baixo**.

Um algoritmo deve obedecer uma **sequência lógica** adequada para cumprir seu papel.

```
x <- 10  
y <- 20  
soma <- x + y
```

correto

```
soma <- x + y  
x <- 10  
y <- 20
```

errado

Palavras finais

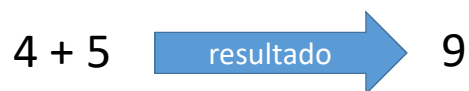
- Nesta seção do curso:
 - Explicações teórico-práticas
 - Excelente para aprendizado e revisão
 - Debug (execução passo a passo)
 - PRÁTICA! Vários exercícios!

Expressões aritméticas

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

Expressões aritméticas



Operadores aritméticos do VisualG

Operador	Significado
+	adição
-	subtração
*	multiplicação
/	divisão
\	divisão inteira
% ou mod	resto da divisão (inteiro)
^	potenciação (real)

Precedência:

- 1º lugar: ^
- 2º lugar: * / \ %
- 3º lugar: + -

Exemplos de expressões aritméticas

$$2 * 6 / 3 \quad \text{Resultado} = 4$$

$$3 + 2 * 4 \quad \text{Resultado} = 11$$

$$(3 + 2) * 4 \quad \text{Resultado} = 20$$

$$2 * 3 ^ 4 \quad \text{Resultado} = 162$$

$$60 / (3 + 2) * 4 \quad \text{Resultado} = 48$$

$$60 / ((3 + 2) * 4) \quad \text{Resultado} = 3$$

Exemplos com o operador "mod" ou %

$$14 \% 3 \quad \text{Resultado} = 2$$

$$19 \% 5 \quad \text{Resultado} = 4$$

Pois:

$$\begin{array}{r|l} 14 & 3 \\ \hline 2 & 4 \end{array}$$

$$\begin{array}{r|l} 19 & 5 \\ \hline 4 & 3 \end{array}$$

Variáveis e tipos básicos em VisualG

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

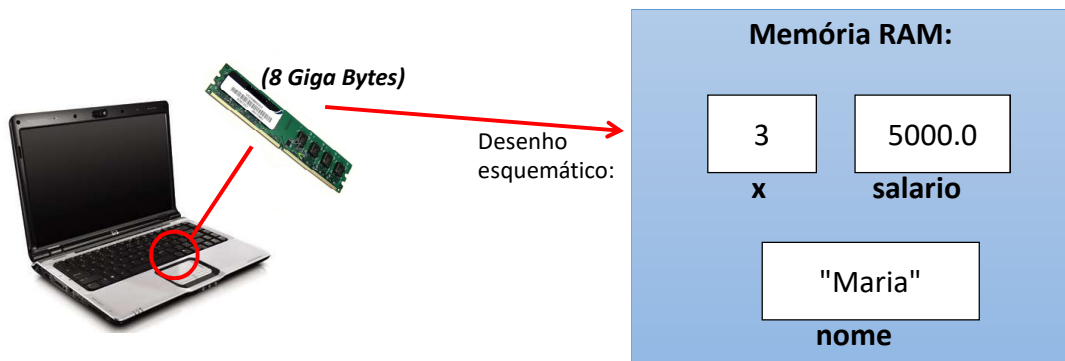
Visão geral

- Um programa de computador em execução lida com dados
- Como esses dados são armazenados?
- Em **VARIÁVEIS!**

Variáveis

Definição informal:

Em programação, uma variável é uma porção de memória (RAM) utilizada para armazenar dados durante a execução dos programas.



Declaração de variáveis

Sintaxe:

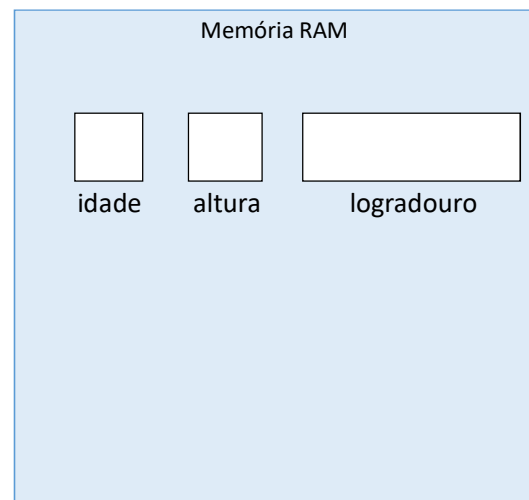
<nome> : <tipo>

Exemplos:

idade : inteiro
altura : real
logradouro : caractere

Uma variável possui:

- Nome (ou identificador)
- Tipo
- Valor
- Endereço

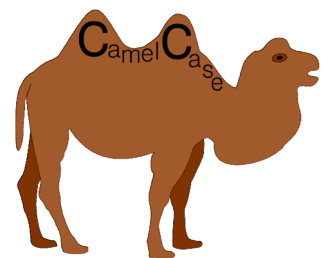


Tipos básicos em VisualG

TIPO	DESCRIÇÃO	VALOR PADRÃO	VALORES POSSÍVEIS
inteiro	número inteiro	0	-2147483648 a 2147483647
real	número com ponto flutuante	0	-1,4024E-37 a 3,4028E+38
caractere	texto	""	(textos)
logico	valor verdade	FALSO	FALSO, VERDADEIRO

Nomes de variáveis

- Não pode começar com dígito: use uma letra ou _
- Não pode ter espaço em branco
- Não usar acentos ou til
- Sugestão: use o padrão "camel case"



Errado:

```
5minutos : inteiro  
salário : inteiro  
salário do funcionário : inteiro
```

Correto:

```
_5minutos : inteiro  
salario : inteiro  
salarioDoFuncionario : inteiro
```


As três operações básicas de programação

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

Um programa de computador é capaz de realizar essencialmente três operações:



Entrada de dados

Usuário → **Programa**
(dentro de variáveis)



Dispositivo de ENTRADA



Também chamada de
LEITURA:

"O programa está lendo dados."

Processamento de dados

É quando o programa realiza os cálculos



O processamento de
dados se dá por um
comando chamado
ATRIBUIÇÃO

```
media <- (x + y) / 2.0
```

Saída de dados

Programa → Usuário



Dispositivo de SAÍDA



Também chamada de
ESCRITA:

"O programa está escrevendo dados."

Saída de dados

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

Saída de dados

Programa → Usuário



Dispositivo de SAÍDA



Também chamada de
ESCRITA:

"O programa está escrevendo dados."

Para escrever na tela um texto qualquer

Sem quebra de linha ao final:

```
escreva("Bom dia!")
```

Com quebra de linha ao final:

```
escreval("Bom dia!")
```

Exemplo

Algoritmo "teste_saida"

Var

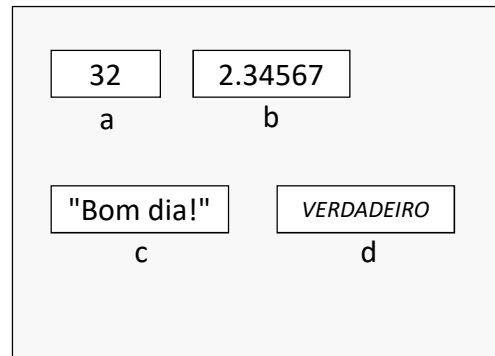
a : inteiro
b : real
c : caractere
d : logico

Inicio

a <- 32
b <- 2.34567
c <- "Bom dia!"
d <- VERDADEIRO

Fimalgoritmo

Na memória:



Exemplo

Algoritmo "teste_saida"

Var

a : inteiro
b : real
c : caractere
d : logico

Inicio

a <- 32
b <- 2.34567
c <- "Bom dia!"
d <- VERDADEIRO

escreval(a)
escreval(b)
escreval(b:10)
escreval(b:10:2)
escreval(c)
escreval(d)

Fimalgoritmo

Escrevendo mais de uma expressão de uma só vez

```
Algoritmo "teste_saida"

Var
    idade : inteiro
    salario : real
    nome : caractere
Inicio
    idade <- 32
    salario <- 4560.9
    nome <- "Maria Silva"

    escreval("A funcionaria ", nome, " ganha ", salario:8:2, " e tem ", idade, " anos.")
Fimalgoritmo
```

Como "saltar" uma linha: use escreval

```
Algoritmo "teste_saida"

Var

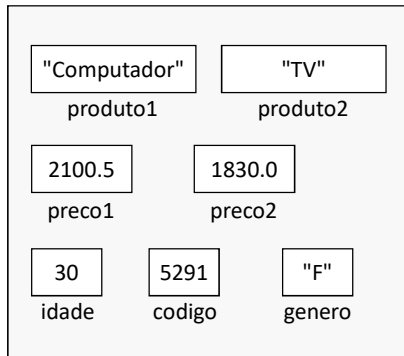
Inicio

    escreval("Bom dia!")
    escreval
    escreval("Boa noite!")

Fimalgoritmo
```

Exercício de fixação

Crie na memória as seguintes variáveis:



Em seguida, usando os valores das variáveis, produza a seguinte saída na tela:

```
Produtos:
0 produto Computador custa R$ 2100.50
0 produto TV custa R$ 1830.00

Codigo = 5291

Dados da pessoa: genero F e idade 30
```

(correção na próxima página)

Algoritmo "exercicio_saida"

Var

`produto1, produto2, genero : caractere`
`preco1, preco2 : real`
`idade, codigo : inteiro`

Inicio

```
produto1 <- "Computador"
produto2 <- "TV"
preco1 <- 2100.5
preco2 <- 1830.0
idade <- 30
codigo <- 5291
genero <- "F"
```

```
escreval("Produtos:")
escreval("0 produto ", produto1, " custa R$ ", preco1:8:2)
escreval("0 produto ", produto2, " custa R$ ", preco2:8:2)
escreval()
escreval("Codigo = ", codigo)
escreval()
escreval("Dados da pessoa: genero ", genero, " e idade ", idade)
```

Fimalgoritmo

Processamento de dados (comando de atribuição)

<http://educandoweb.com.br>


Prof. Dr. Nelio Alves

Processamento de dados

Comando de atribuição.

Sintaxe:

<variável> <- <expressão>


Lê-se "recebe"

REGRA:

- 1) A expressão é calculada
- 2) O resultado da expressão é armazenado na variável

Exemplo 1

`x, y : inteiro`

`x <- 5`

`y <- 2 * x`

`escreval(x)`

`escreval(y)`

Exemplo 2

`x : inteiro`

`y : real`

`x <- 5`

`y <- 2 * x`

`escreval(x)`

`escreval(y)`

Exemplo 3

b1, b2, h, area : real

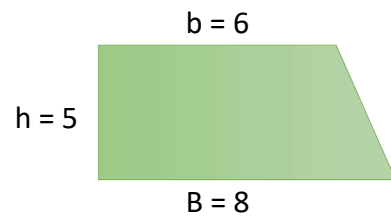
b1 <- 6.0

b2 <- 8.0

h <- 5.0

area <- (b1 + b2) / 2.0 * h;

escreval(area)



$$area = \frac{(b + B)}{2} \times h$$

No exemplo:

$$\begin{aligned} area &= \frac{(6 + 8)}{2} \times 5 \\ &= \frac{14}{2} \times 5 = 7 \times 5 = 35 \end{aligned}$$

Exemplo 4

a, b, resultado : inteiro

a <- 5

b <- 2

resultado <- a / b **// ERRO!**

escreval(resultado)

SOLUÇÃO:

resultado <- a \ b

Exemplo 5

```
a : real  
b : inteiro
```

```
a <- 5.0  
b <- a      // ERRO
```

```
escreval(b)
```

SOLUÇÃO:

```
b <- Int(a)
```

Entrada de dados

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

Entrada de dados

Usuário → **Programa**
(dentro de variáveis)



Dispositivo de ENTRADA

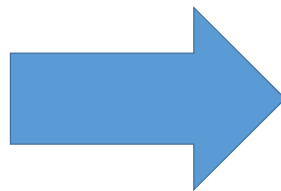


Também chamada de
LEITURA:

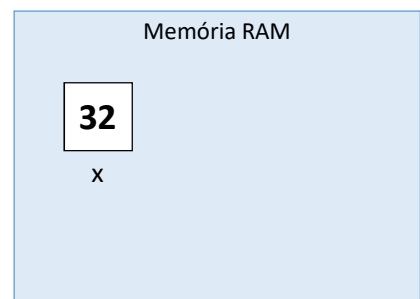
"O programa está lendo dados."



32 ENTER



`x : inteiro`



Comando "leia"

Para fazer o programa ler um valor e armazená-lo em uma variável, usa-se o comando "leia":

`leia(x)`

Exemplo

Algoritmo "teste_entrada"

Var

idade : inteiro
salario1, salario2 : real
nome : caractere

Inicio

escreva("Digite seu nome: ")
leia(nome)
escreva("Digite sua idade: ")
leia(idade)
escreva("Digite quanto voce ganhou nos dois ultimos meses:")
leia(salario1)
leia(salario2)

escreval("DADOS DIGITADOS:")
escreval("Nome = ", nome)
escreval("Idade = ", idade)
escreval("Salario 1 = ", salario1:8:2)
escreval("Salario 2 = ", salario2:8:2)

Fimalgoritmo

Funções matemáticas

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

Algumas funções matemáticas

Exemplo	Significado
<code>A <- RaizQ(x)</code>	Variável A recebe a raiz quadrada de x
<code>A <- Exp(x, y)</code>	Variável A recebe o resultado de x elevado a y
<code>A <- Pi</code>	Variável A recebe o valor de Pi
<code>A <- Abs(x)</code>	Variável A recebe o valor absoluto de x

Lista completa:

https://manual.visualg3.com.br/doku.php?id=manual#as_funcoes_do_visualg_versao_30

Algoritmo "teste_entrada"

Var

x, y, z, a, b, c, d : real

Inicio

x <- 2.0

y <- 3.0

z <- -7.0

a <- Exp(x, y)

b <- RaizQ(y)

c <- Pi

d <- Abs(z)

escreval(a)

escreval(b)

escreval(c)

escreval(d)

Fimalgoritmo

Incluindo funções em expressões maiores

$$x = \frac{-b \pm \sqrt{\Delta}}{2.a}$$

$$\Delta = b^2 - 4ac$$

delta <- Exp(b, 2.0) - 4 * a * c

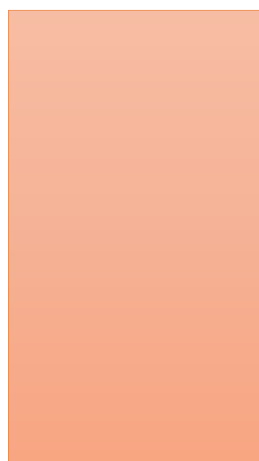
x1 <- (-b + RaizQ(delta)) / (2.0 * a)

x2 <- (-b - RaizQ(delta)) / (2.0 * a)

Discussão do exercício "terreno"

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves



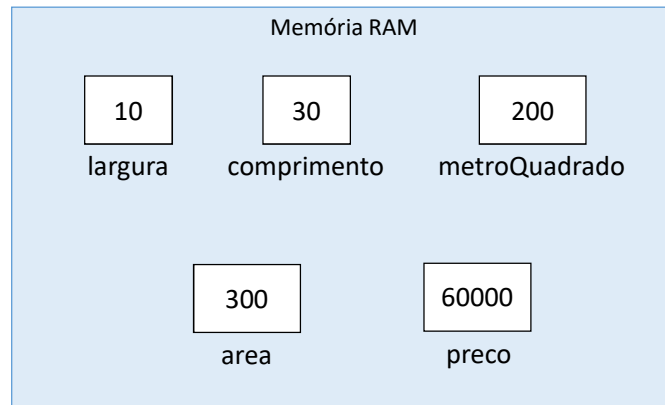
30.0 metros

10.0 metros

$$\text{área} = \text{largura} \times \text{comprimento}$$

$$\text{preço} = \text{área} \times \text{preço do metro quadrado}$$

Digite a largura do terreno: **10.0**
Digite o comprimento do terreno: **30.0**
Digite o valor do metro quadrado: **200.00**
Area do terreno = 300.00
Preco do terreno = 60000.00



Algoritmo "terreno"

Var

largura, comprimento, metroQuadrado : real
area, preco : real

Inicio

escreva("Digite a largura do terreno: ")
leia(largura)
escreva("Digite o comprimento do terreno: ")
leia(comprimento)
escreva("Digite o valor do metro quadrado: ")
leia(metroQuadrado)

area <- largura * comprimento
preco <- area * metroQuadrado

escreval("Area do terreno = ", area:10:2)
escreval("Preco do terreno = ", preco:10:2)

Fimalgoritmo

Como utilizar o debug no VisualG (execução passo a passo)

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

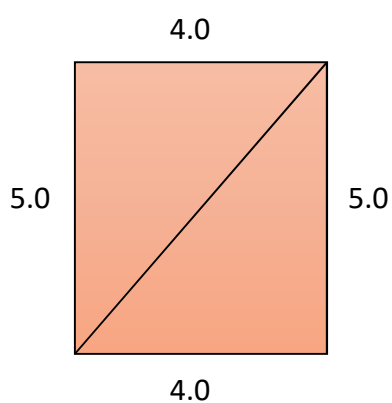
Como executar o debug no VisualG

- Iniciar o debug: F8
- Parar a execução: CTRL + F2
- Marcar/desmarcar uma linha de breakpoint: F5

Discussão do exercício "retângulo"

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves



$$\text{área} = \text{base} \times \text{altura}$$

$$\text{perímetro} = 2 \times \text{base} + 2 \times \text{altura}$$

$$\text{diagonal} = \sqrt{\text{base}^2 + \text{altura}^2}$$

Solução dos exercícios:

github.com/acenelio/curso-algoritmos