



## Formação Desenvolvedor Moderno Módulo: Lógica de Programação

Capítulo: Estrutura Sequencial

<https://devsuperior.com.br>

1

## Visão geral sobre o capítulo

<https://devsuperior.com.br>

Prof. Dr. Nelio Alves

2

## Conteúdo

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

3

## 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

4

## Palavras finais

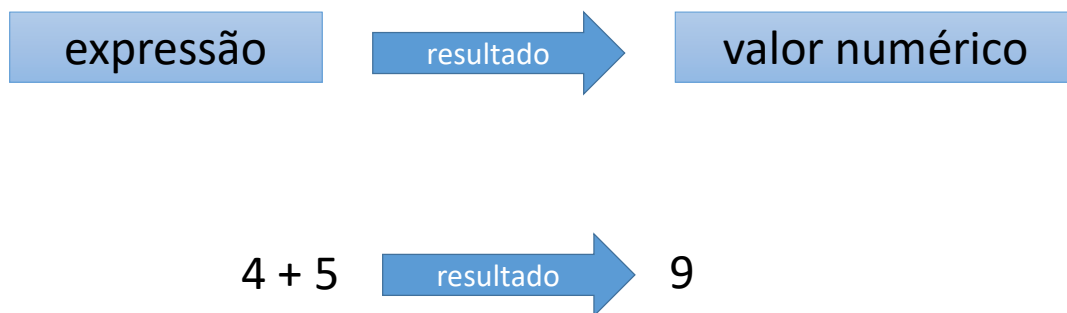
- Nesta seção do curso:
  - Explicações teórico-práticas
    - Excelente para aprendizado e revisão
  - Debug (execução passo a passo)
  - PRÁTICA! Vários exercícios!

## Expressões aritméticas

<https://devsuperior.com.br>

Prof. Dr. Nelio Alves

## Expressões aritméticas



7

## 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: + -

8

## Exemplos de expressões aritméticas

$2 * 6 / 3$  Resultado = 4

$3 + 2 * 4$  Resultado = 11

$(3 + 2) * 4$  Resultado = 20

$2 * 3 ^ 4$  Resultado = 162

$60 / (3 + 2) * 4$  Resultado = 48

$60 / ((3 + 2) * 4)$  Resultado = 3

9

## Exemplos com o operador "mod" ou %

$14 \% 3$  Resultado = 2

$19 \% 5$  Resultado = 4

**Pois:**

14		3	
2			4

19		5	
4			3

10

# Variáveis e tipos básicos em VisualG

<https://devsuperior.com.br>

Prof. Dr. Nelio Alves

11

## Visão geral

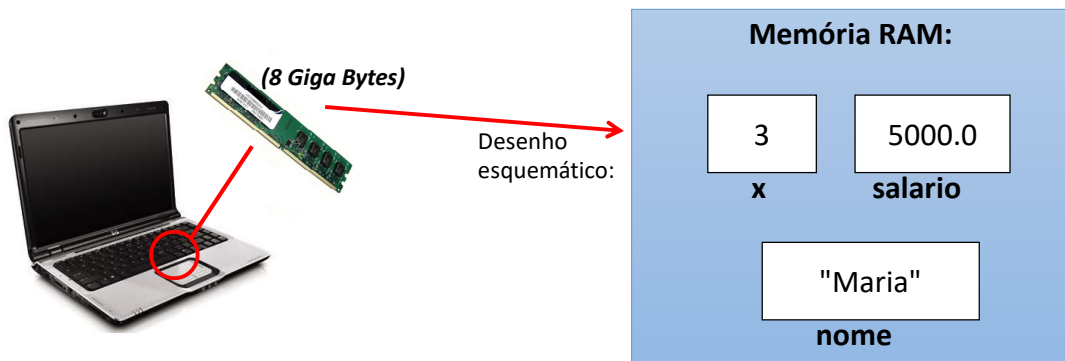
- Um programa de computador em execução lida com dados
- Como esses dados são armazenados?
- Em **VARIÁVEIS!**

12

# Variáveis

## Definição informal:

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



13

## Declaração de variáveis

### Sintaxe:

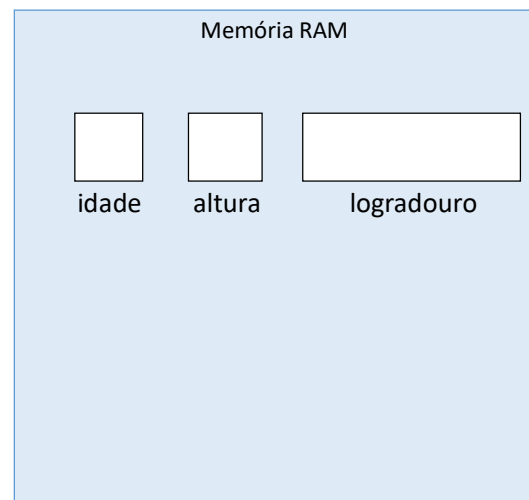
**<nome> : <tipo>**

### Exemplos:

idade : inteiro  
altura : real  
logradouro : caractere

### Uma variável possui:

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



14

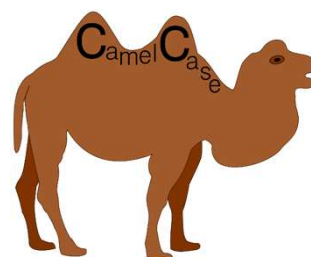
## Tipos básicos em VisualG

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

15

## 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
```

16



# As três operações básicas de programação

<https://devsuperior.com.br>

Prof. Dr. Nelio Alves

17

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



18

## Entrada de dados

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



Dispositivo de ENTRADA



Também chamada de  
**LEITURA:**

"O programa está lendo dados."

19

## 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
```

20

## Saída de dados

**Programa → Usuário**



Dispositivo de SAÍDA



Também chamada de  
**ESCRITA:**

"O programa está escrevendo dados."

## Saída de dados

<https://devsuperior.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

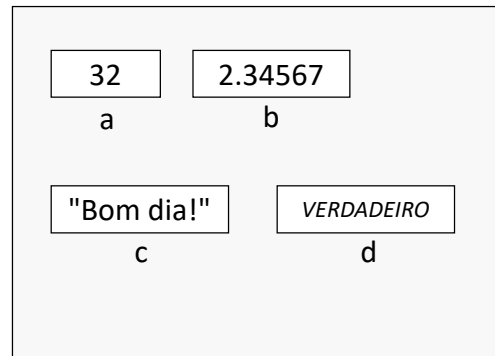
a : inteiro  
b : real  
c : caractere  
d : logico

Inicio

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

Fimalgoritmo

Na memória:



25

## 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

26

## 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
```

27

## Como "saltar" uma linha: use escreval

```
Algoritmo "teste_saida"

Var

Inicio

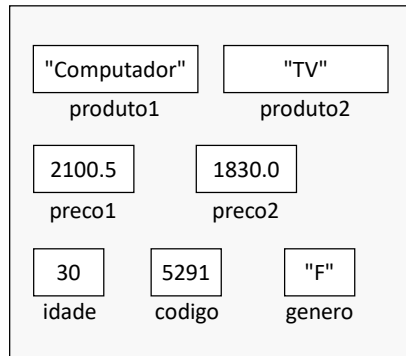
    escreval("Bom dia!")
    escreval
    escreval("Boa noite!")

Fimalgoritmo
```

28

## Exercício de fixação

Crie na memória as seguintes variáveis:



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

```
Produtos:  
O produto Computador custa R$ 2100.50  
O produto TV custa R$ 1830.00  
  
Codigo = 5291  
  
Dados da pessoa: genero F e idade 30
```

(correção na próxima página)

29

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

Fimalgoritmo

30

# Processamento de dados (comando de atribuição)

<https://devsuperior.com.br>

Prof. Dr. Nelio Alves

31

## 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

32



## Exemplo 1

`x, y : inteiro`

`x <- 5`

`y <- 2 * x`

`escreval(x)`

`escreval(y)`

33

## Exemplo 2

`x : inteiro`

`y : real`

`x <- 5`

`y <- 2 * x`

`escreval(x)`

`escreval(y)`

34

### Exemplo 3

b1, b2, h, area : real

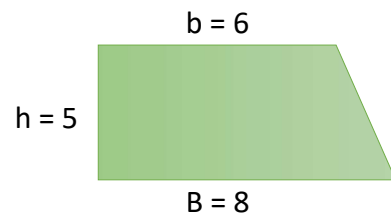
```
b1 <- 6.0
```

```
b2 <- 8.0
```

```
h <- 5.0
```

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

```
escreval(area)
```



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

**No exemplo:**

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

35

### Exemplo 4

a, b, resultado : inteiro

```
a <- 5
```

```
b <- 2
```

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

```
escreval(resultado)
```

**SOLUÇÃO:**

```
resultado <- a \ b
```

36

## Exemplo 5

```
a : real  
b : inteiro  
  
a <- 5.0  
b <- a      // ERRO  
  
escreval(b)
```

SOLUÇÃO:

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

37

## Entrada de dados

<https://devsuperior.com.br>

Prof. Dr. Nelio Alves

38

## Entrada de dados

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



Dispositivo de ENTRADA



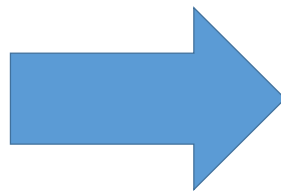
Também chamada de  
**LEITURA:**

"O programa está lendo dados."

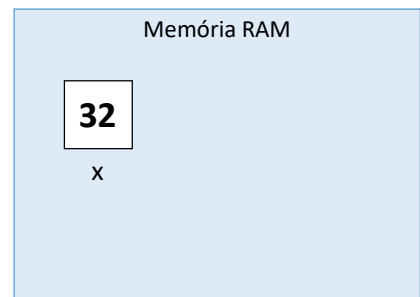
39



32      ENTER



`x : inteiro`



40

## Comando "leia"

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

`leia(x)`

41

## 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

42

# Funções matemáticas

<https://devsuperior.com.br>

Prof. Dr. Nelio Alves

43

## Algumas funções matemáticas

Exemplo	Significado
<code>A &lt;- RaizQ(x)</code>	Variável A recebe a raiz quadrada de x
<code>A &lt;- Exp(x, y)</code>	Variável A recebe o resultado de x elevado a y
<code>A &lt;- Pi</code>	Variável A recebe o valor de Pi
<code>A &lt;- 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](https://manual.visualg3.com.br/doku.php?id=manual#as_funcoes_do_visualg_versao_30)

44

```

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

```

45

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)
```

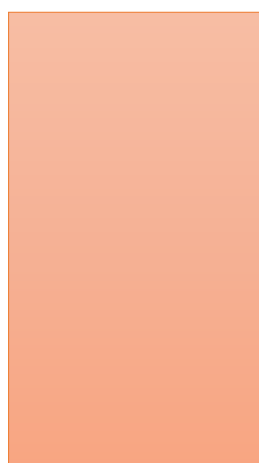
46

# Discussão do exercício "terreno"

<https://devsuperior.com.br>

Prof. Dr. Nelio Alves

47



30.0 metros

10.0 metros

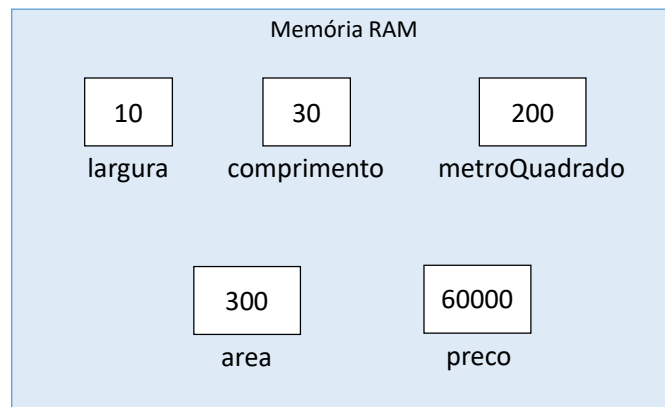
$$\text{área} = \text{largura} \times \text{comprimento}$$

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

48



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



49

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

50

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

<https://devsuperior.com.br>

Prof. Dr. Nelio Alves

51

## Como executar o debug no VisualG

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

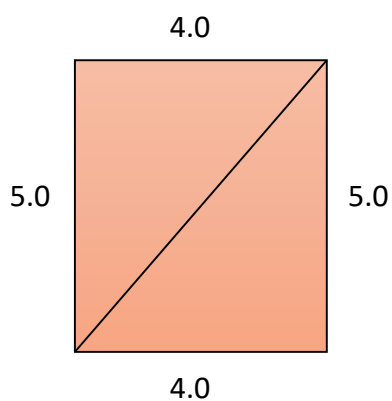
52

# Discussão do exercício "retangulo"

<https://devsuperior.com.br>

Prof. Dr. Nelio Alves

53



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

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

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

54

Hora de praticar!

Resolver os demais problemas da lista de exercícios deste capítulo.

**Soluções dos exercícios:**

[github.com/acenelio/curso-algoritmos](https://github.com/acenelio/curso-algoritmos)