

ALGORITMO



Algoritmo

Sequência finita de instruções para se resolver um problema.

Sequência de **passos lógicos** que visam atingir um determinado **objetivo**.

Mas o que algoritmo e
automação tem a ver com
programação de computadores?

Computador

- Hardware - parte física (a máquina em si)
- Software - parte lógica (programas)
 - Sistema operacional (Windows, Linux, Mac)
 - Aplicativos (apps de escritório, app de câmera, navegador web)
 - Jogos
 - Utilitários (Antivírus, compactador de arquivos)
 - Outros



O que é preciso para se fazer um programa de computador?

Vamos precisar de:

- Uma **linguagem de programação**: regras **léxicas** e **sintáticas** para se escrever o programa
- Uma **IDE**: software para editar e testar o programa

Linguagem de programação

É um conjunto de regras **léxicas** (ortografia) e **sintáticas** (gramática) para se escrever programas.

Léxica

Diz respeito à correção das **palavras** "isoladas" (ortografia).

Exemplo (Português):

cachorro

caxorro



Linguagem de programação:

main

maim



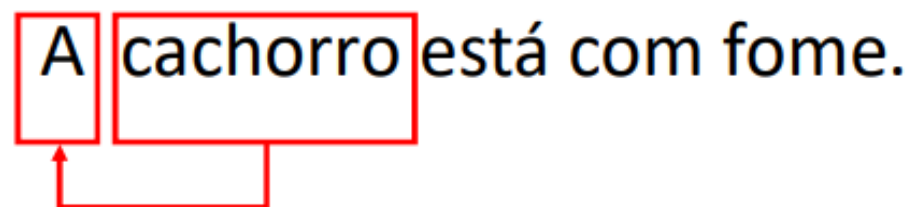
Sintática

Diz respeito à correção das **sentenças** (gramática).

Exemplo (Português):

O cachorro está com fome.

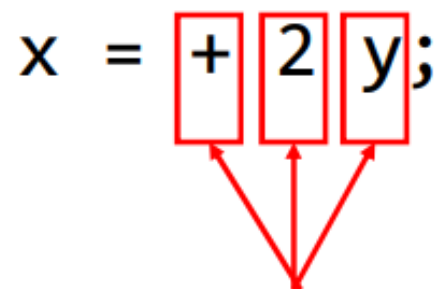
A cachorro está com fome.



Linguagem de programação:

$x = 2 + y;$

$x = + 2 y;$



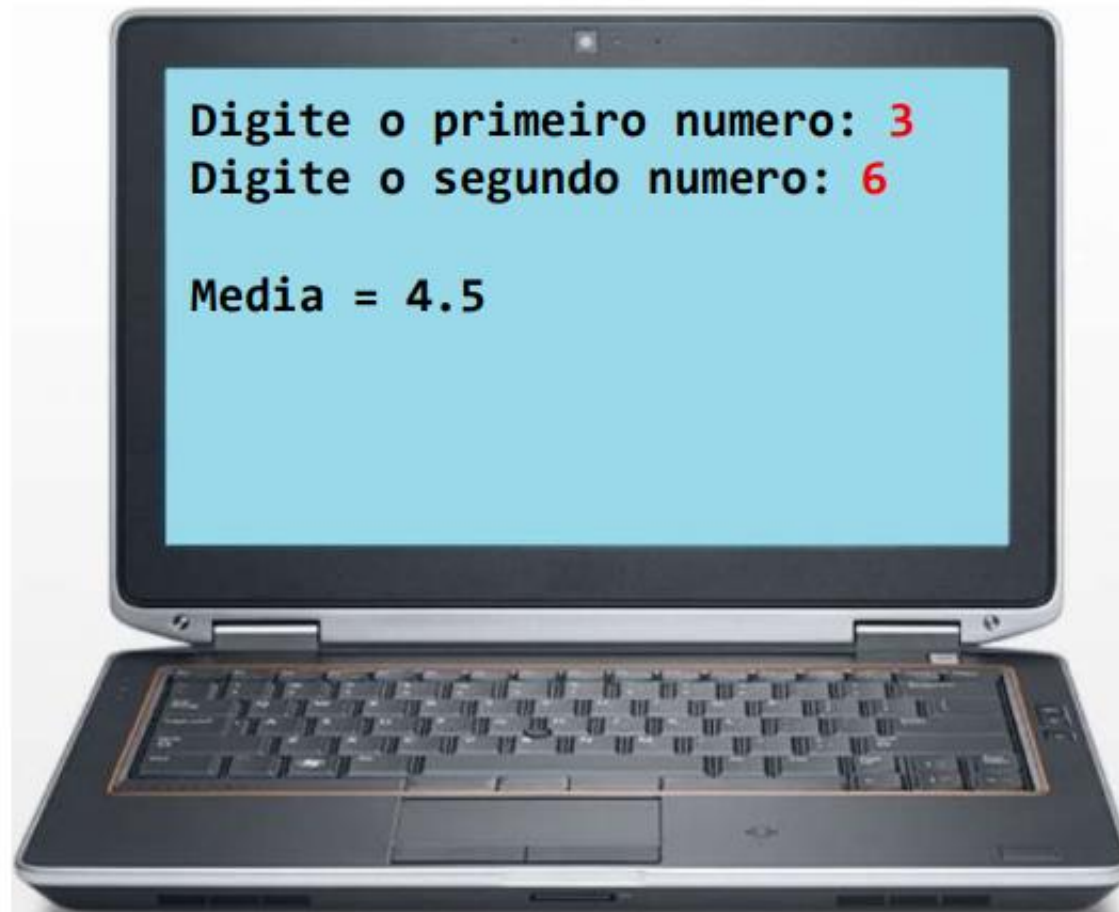
Linguagem de programação

Exemplos de linguagens de programação:

C, Pascal, C++, Java, C#, Python, Ruby, PHP, JavaScript, etc.

Exemplo de um programa:

Suponha um programa que solicita do usuário dois números e depois mostra a média aritmética deles:



Solução em linguagem C

```
#include <stdio.h>

int main() {
    double x, y, media;

    printf("Digite o primeiro numero: ");
    scanf("%lf", &x);
    printf("Digite o segundo numero: ");
    scanf("%lf", &y);
    media = (x + y) / 2.0;
    printf("Media = %.1f\n", media);
    return 0;
}
```

Solução em linguagem C++

```
#include <iostream>

using namespace std;

int main() {
    double x, y, media;

    cout << "Digite o primeiro numero: ";
    cin >> x;
    cout << "Digite o segundo numero: ";
    cin >> y;
    media = (x + y) / 2.0;
    cout << "Media = " << media << endl;
    return 0;
}
```

Solução em linguagem C#

```
using System;

namespace programa {
    class Program {
        static void Main(string[] args) {
            double x, y, media;

            Console.Write("Digite o primeiro numero: ");
            x = double.Parse(Console.ReadLine());
            Console.Write("Digite o segundo numero: ");
            y = double.Parse(Console.ReadLine());
            media = (x + y) / 2.0;
            Console.WriteLine("Media = " + media);
        }
    }
}
```


Solução em linguagem Java

```
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        double x, y, media;

        System.out.print("Digite o primeiro numero: ");
        x = sc.nextDouble();
        System.out.print("Digite o segundo numero: ");
        y = sc.nextDouble();
        media = (x + y) / 2.0;
        System.out.println("Media = " + media);
        sc.close();
    }
}
```

IDE: Ambiente de Desenvolvimento Integrado

IDE – Ambiente Integrado de Desenvolvimento

É um conjunto de softwares utilizado para a construção de programas.

Exemplos:

C/C++ : **Code Blocks**

Java : **Eclipse, NetBeans**

C# : **Microsoft Visual Studio**

Funcionalidades de uma IDE

- Edição de código fonte (indentação, autocompletar, destaque de palavras, etc.)
- Depuração e testes
- Construção do produto final (build)
- Sugestão de modelos (templates)
- Auxiliar em várias tarefas do seu projeto
- Etc.

A linguagem "Portugol" e o VisualG

Linguagem "Portugol"

- Linguagem didática, simplificada, com objetivo didático para estudantes de língua portuguesa
- Usada por autores de livros e professores há décadas

Exemplo linguagem C:

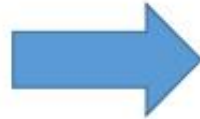
```
double x, y, media;  
  
printf("Digite o primeiro numero: ");  
scanf("%lf", &x);  
printf("Digite o segundo numero: ");  
scanf("%lf", &y);  
media = (x + y) / 2.0;  
printf("Media = %.1f\n", media);
```

Exemplo linguagem Portugol:

```
x, y, media : real  
  
escreva("Digite o primeiro numero: ")  
leia(x)  
escreva("Digite o segundo numero: ")  
leia(y)  
media <- (x + y) / 2  
escreva("Media = ", media)
```

Aprender a lógica

```
x, y, media : real  
  
escreva("Digite o primeiro numero: ")  
leia(x)  
escreva("Digite o segundo numero: ")  
leia(y)  
media <- (x + y) / 2  
escreva("Media = ", media)
```



Aprender a implementar a lógica na linguagem X

```
double x, y, media;  
  
printf("Digite o primeiro numero: ");  
scanf("%lf", &x);  
printf("Digite o segundo numero: ");  
scanf("%lf", &y);  
media = (x + y) / 2.0;  
printf("Media = %.1f\n", media);
```

Portugol possui "dialetos"

- A sintaxe do "Portugol" varia um pouco de autor para autor (dialetos)
- A linguagem às vezes é referida por outro nome ("Português estruturado", "Linguagem algorítmica", etc.)

```
x, y, media : real  
  
escreva("Digite o primeiro numero: ")  
leia(x)  
escreva("Digite o segundo numero: ")  
leia(y)  
media <- (x + y) / 2  
escreva("Media = ", media)
```

Seu primeiro algoritmo escrito na
linguagem "Portugol", usando
VisualG

Algoritmo "primeiro"



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.

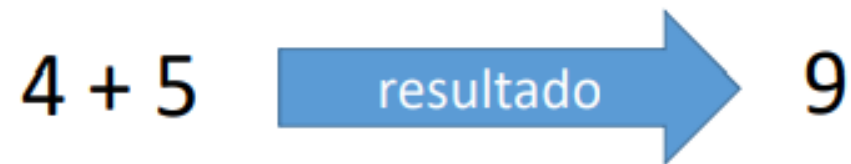
```
x <- 10  
y <- 20  
soma <- x + y
```

correto

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

errado

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

$$2 * 6 / 3$$

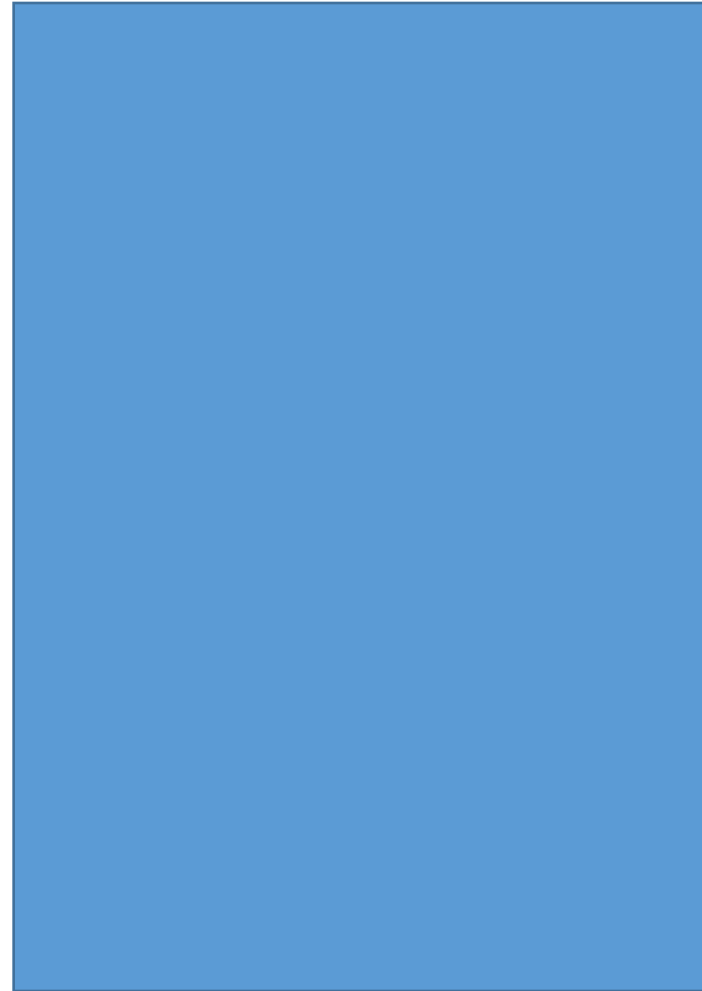
$$3 + 2 * 4$$

$$(3 + 2) * 4$$

$$2 * 3 ^ 4$$

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

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



Exemplos com o operador "mod" ou %

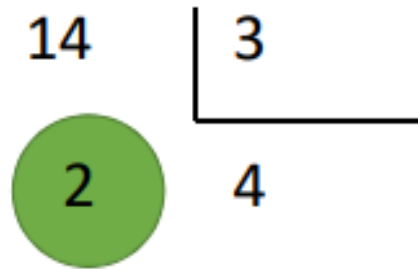
$$14 \% 3$$

Resultado = 2

$$19 \% 5$$

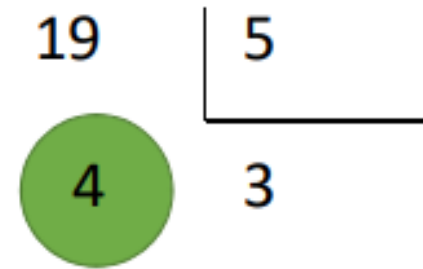
Resultado = 4

Pois:



A handwritten division problem: 14 divided by 3. The quotient is 4 and the remainder is 2. The remainder '2' is circled in green.

$$\begin{array}{r} 14 \quad | \quad 3 \\ 4 \quad | \quad 12 \\ \hline 2 \end{array}$$



A handwritten division problem: 19 divided by 5. The quotient is 3 and the remainder is 4. The remainder '4' is circled in green.

$$\begin{array}{r} 19 \quad | \quad 5 \\ 3 \quad | \quad 15 \\ \hline 4 \end{array}$$

Variáveis e tipos básicos em VisualG

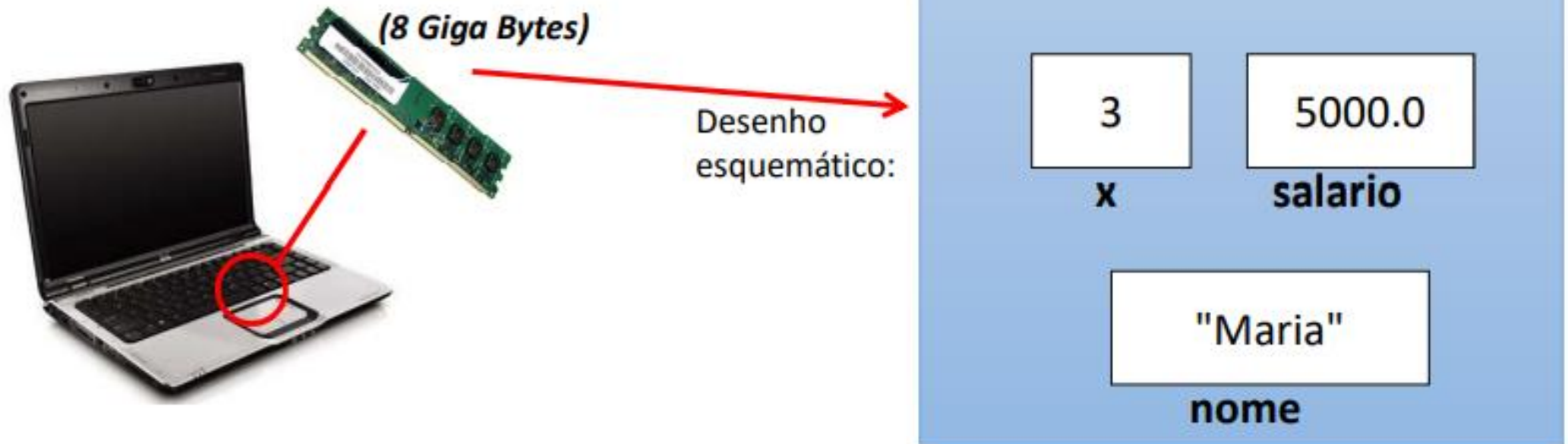
Visão geral

- 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:

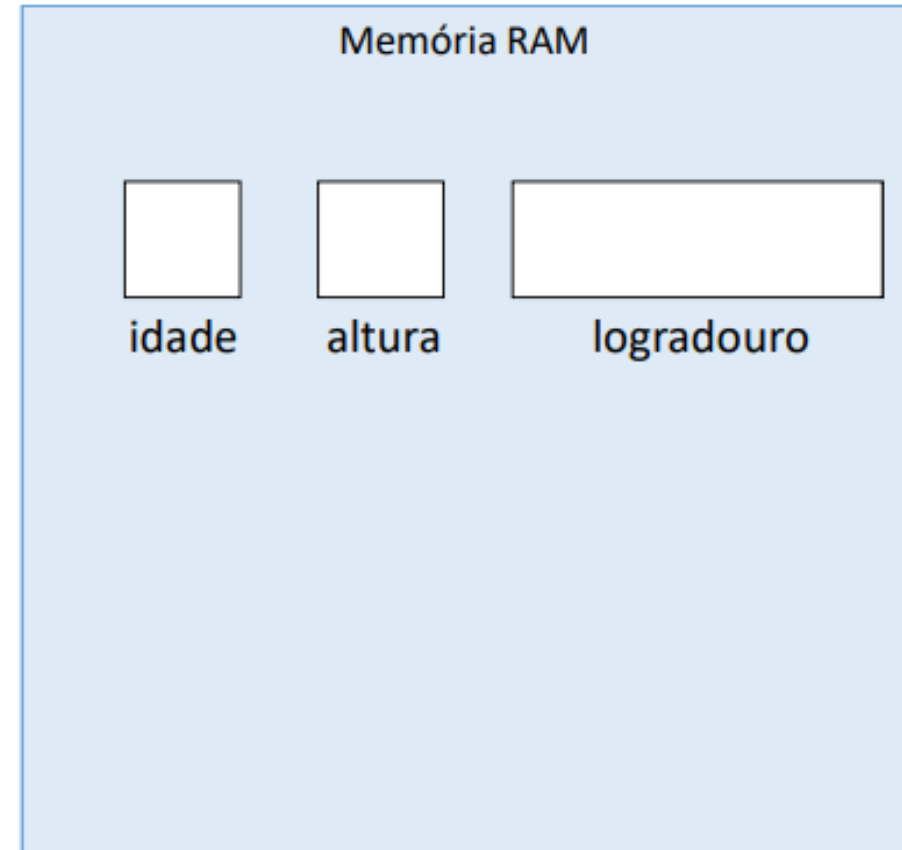
`idade : inteiro`

`altura : real`

`logradouro : caractere`

Uma variável possui:

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

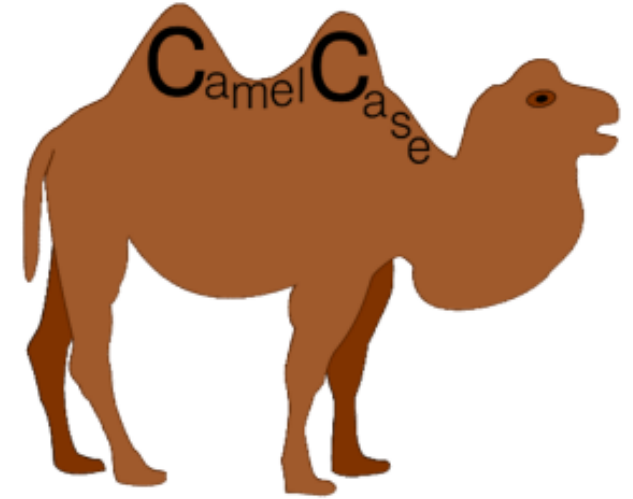


Tipos básicos em VisualG

| 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

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

Entrada
de dados



Processamento
de dados



Saída
de dados



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

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"

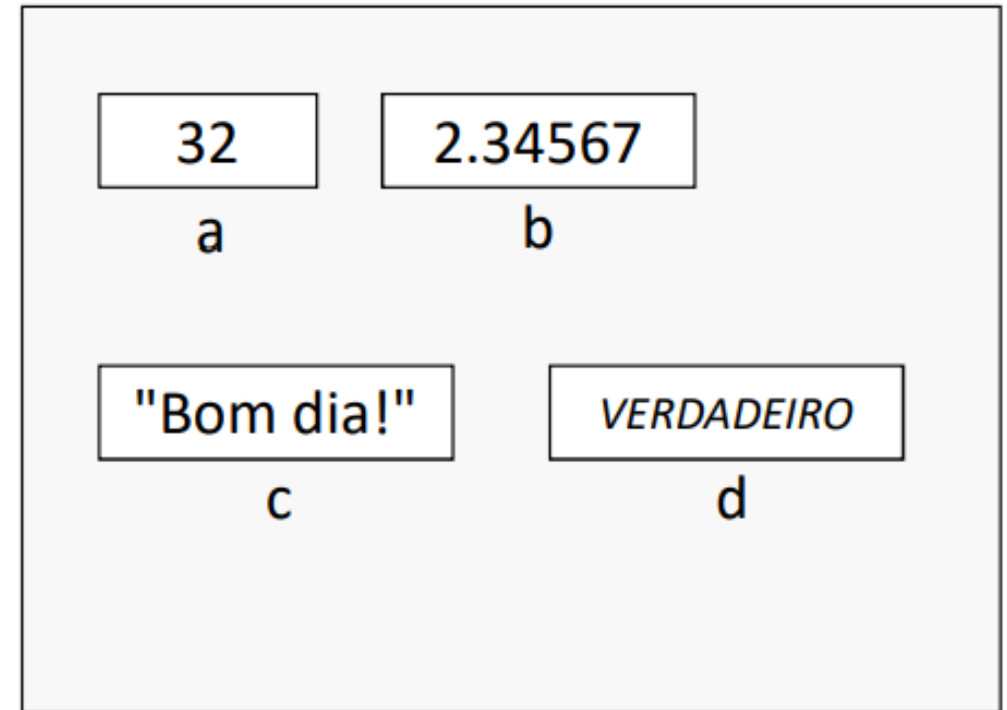
Var

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

Inicio

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

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

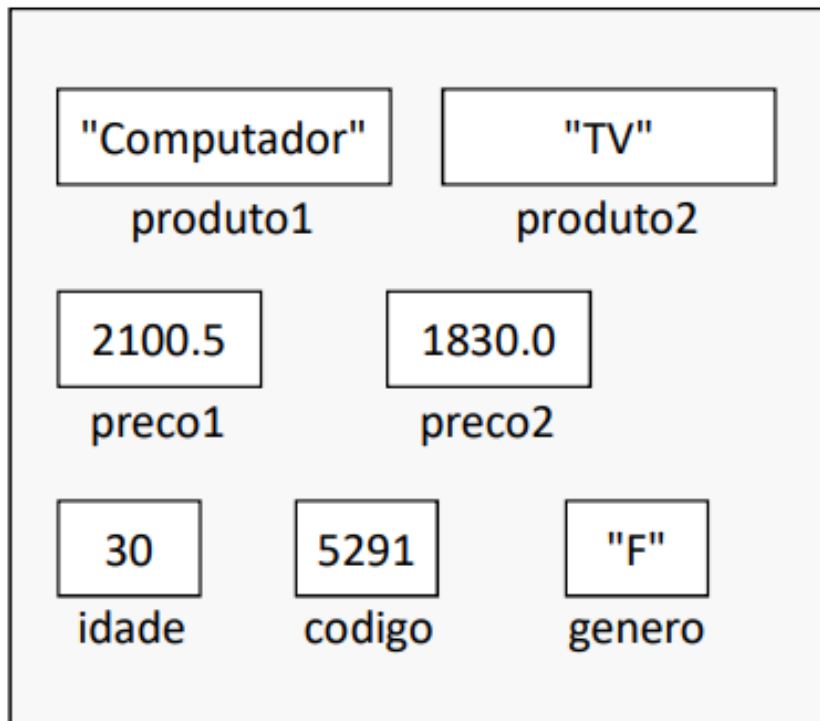
Inicio

```
    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


Processamento de dados
(comando de atribuição)

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
```

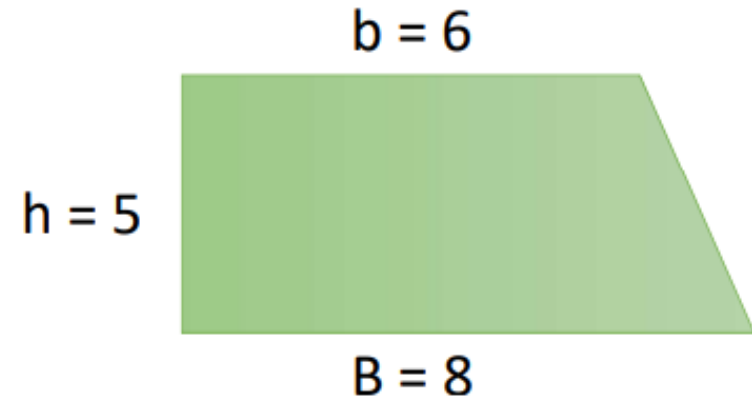
```
b1 <- 6.0
```

```
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

Entrada de dados

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



Dispositivo de ENTRADA



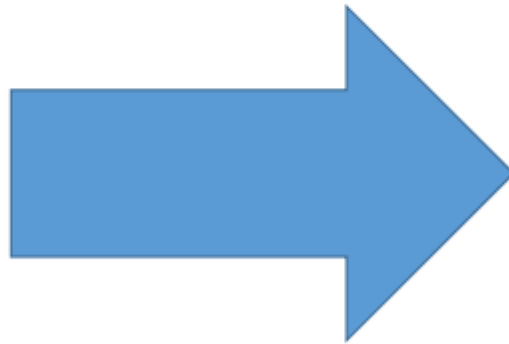
Também chamada de
LEITURA:

"O programa está lendo dados."

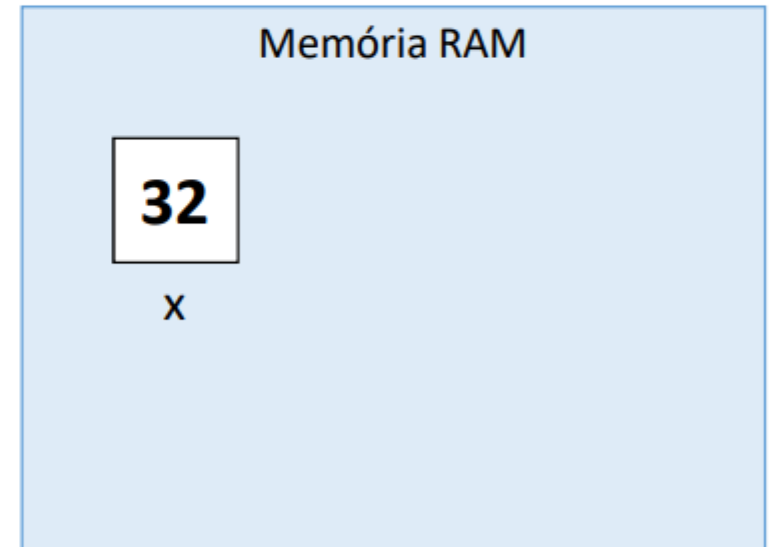


32

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)
escreval("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

Problema "soma"

Fazer um programa para ler dois valores inteiros X e Y, e depois mostrar na tela o valor da soma destes números.

Exemplo 1:

```
Digite o valor de X: 8  
Digite o valor de Y: 10  
SOMA = 18
```

Exemplo 2:

```
Digite o valor de X: 12  
Digite o valor de Y: 31  
SOMA = 43
```

Problema "pagamento"

Fazer um programa para ler o nome de um(a) funcionário(a), o valor que ele(a) recebe por hora, e a quantidade de horas trabalhadas por ele(a). Ao final, mostrar o valor do pagamento do funcionário com uma mensagem explicativa, conforme exemplo.

Exemplo 1:

```
Nome: Joao Silva  
Valor por hora: 50.00  
Horas trabalhadas: 60  
O pagamento para Joao Silva deve ser 3000.00
```

Exemplo 2:

```
Nome: Maria Dias  
Valor por hora: 60.00  
Horas trabalhadas: 100  
O pagamento para Maria Dias deve ser 6000.00
```

Problema "idades"

Fazer um programa para ler o nome e idade de duas pessoas. Ao final mostrar uma mensagem com os nomes e a idade média entre essas pessoas, com uma casa decimal, conforme exemplo.

Exemplo:

Dados da primeira pessoa:

Nome: **Maria Silva**

Idade: **19**

Dados da segunda pessoa:

Nome: **Joao Melo**

Idade **20**

A idade média de Maria Silva e Joao Melo é de 19.5 anos

Problema "troco"

Fazer um programa para calcular o troco no processo de pagamento de um produto de uma mercearia. O programa deve ler o preço unitário do produto, a quantidade de unidades compradas deste produto, e o valor em dinheiro dado pelo cliente (suponha que haja dinheiro suficiente). Seu programa deve mostrar o valor do troco a ser devolvido ao cliente.

Exemplo 1:

```
Preço unitário do produto: 8.00  
Quantidade comprada: 2  
Dinheiro recebido: 20.00  
TROCO = 4.00
```

Exemplo 2:

```
Preço unitário do produto: 30.00  
Quantidade comprada: 3  
Dinheiro recebido: 100.00  
TROCO = 10.00
```

Problema "circulo"

Fazer um programa para ler o valor "r" do raio de um círculo, e depois mostrar o valor da área do círculo com três casas decimais. A fórmula da área do círculo é a seguinte: $area = \pi \cdot r^2$. Você pode usar o valor de π fornecido pela biblioteca da sua linguagem de programação, ou então, se preferir, use diretamente o valor 3.14159.

Exemplo 1:

```
Digite o valor do raio do circulo: 2.0  
AREA = 12.566
```

Exemplo 2:

```
Digite o valor do raio do circulo: 13.2  
AREA = 547.391
```


Problema "consumo"

Fazer um programa para ler a distância total (em km) percorrida por um carro, bem como o total de combustível gasto por este carro ao percorrer tal distância. Seu programa deve mostrar o consumo médio do carro, com três casas decimais.

Exemplo 1:

```
Distancia percorrida: 500  
Combustível gasto: 38.5  
Consumo medio = 12.987
```

Exemplo 2:

```
Distancia percorrida: 1108  
Combustível gasto: 71.4  
Consumo medio = 15.518
```

Problema "terreno"

Fazer um programa para ler as medidas da largura e comprimento de um terreno retangular com uma casa decimal, bem como o valor do metro quadrado do terreno com duas casas decimais. Em seguida, o programa deve mostrar o valor da área do terreno, bem como o valor do preço do terreno, ambos com duas casas decimais, conforme exemplo.

Exemplo 1:

```
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
```

Exemplo 2:

```
Digite a largura do terreno: 12.0
Digite o comprimento do terreno: 20.0
Digite o valor do metro quadrado: 150.00
Area do terreno = 240.00
Preco do terreno = 36000.00
```

Faça um algoritmo que leia 5 números e mostre:

- Cada número multiplicado por 10
- A soma dos 5 números
- A média dos 5 números

Desafio 2

Faça um algoritmo que leia o **Nome, Sobrenome, Peso, Altura e Idade**

Mostre:

- Nome e Sobrenome na mesma linha
- O IMC da pessoa
- O peso da pessoa na Terra, em Marte e em Saturno
- A idade da pessoa em Meses e em dias

Desafio 3

Faça um algoritmo que leia o **salário e mostre**

Mostre:

- A diferença do salário digitado para o salário mínimo
- O salário multiplicado por 10
- Metade do salário
- O salário com 10% de aumento
- O salário com 34% de aumento