

INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
RIO GRANDE DO NORTE




# ALGORITMOS

Expressões e Instruções Primitivas

**Alessandro J. de Souza**

DIATINF - IFRN

 [alessandro.souza@ifrn.edu.br](mailto:alessandro.souza@ifrn.edu.br)

 [twitter.com/ajdsouza](https://twitter.com/ajdsouza)



# Agenda

## ✧ Expressões

- Conceitos
- Operadores
- Tipos de Expressões
- Avaliação de Expressões

## ✧ Instruções Primitivas

- Instrução Primitiva de atribuição
- Instrução Primitiva de saída de dados
- Instrução Primitiva de entrada de dados



# Expressões

## ✧ Conceito

- O conceito de **expressão** em termos computacionais está intimamente ligado ao conceito de expressão (**ou fórmula**) matemática, onde um conjunto de variáveis e constantes numéricas **relacionam-se** por meio de **operadores aritméticos** para resultar num valor.

$AREA \leftarrow (BASE * ALTURA) / 2$



# Expressões

## ✧ Operadores

- **Operadores** são elementos funcionais que atuam sobre **operandos** e produzem um determinado resultado.
- Os OPERADORES podem ser classificados em:
  - **binários** : “+”, “-”, “\*”, “/”, “MOD” (ou %) e “^”
  - **unários** : “+” e “-”
  - **Relacionais** : “=”, “<”, “>”, “<>” (diferente), “>=”(maior igual), “<=”(menor igual)



# Expressões

## ✧ Tipos de Expressões

- **Expressões Aritméticas:** são aquelas cujo **resultado da avaliação é do tipo numérico**, seja ele inteiro ou real.
  - Somente o uso de operadores aritméticos e variáveis numéricas é permitido em expressões deste tipo.

**Tabela 5.1** Operadores aritméticos e sua ordem de prioridade.

Operador	Tipo	Operação	Prioridade
+	Binário	Adição	4
-	Binário	Subtração	4
*	Binário	Multiplificação	3
/	Binário	Divisão inteira	3
^	Binário	Exponenciação	2
+	Unário	Manutenção de sinal	1
-	Unário	Inversão de sinal	1



# Expressões

## ✧ Tipos de Expressões

- **Expressões Lógicas**: são aquelas cujo **resultado da avaliação é um valor lógico** (*verdadeiro* ou *falso*).

**Tabela 5.2** Operadores lógicos e suas relações de prioridade.

Operador	Tipo	Operação	Prioridade
OU	Binário	<b>Disjunção</b> (resulta VERDADEIRO se um dos operandos for verdadeiro)	3
E	Binário	<b>Conjunção</b> (resulta VERDADEIRO se os dois operandos forem verdadeiro)	2
NAO	Unário	<b>Negação</b> (Ex.: nao VERDADEIRO = FALSO )	1



# Expressões

## ✧ Tipos de Expressões

- *Expressões Literais*: são aquelas cujo **resultado da avaliação é um valor literal**.
  - Os tipos de operadores existentes variam de uma linguagem de programação para outra, não havendo uma padronização.
  - Em VISUALG o operador “+” faz a operação entre literais

**"REFRIGERA" + "DOR" e o resultado de sua avaliação é "REFRIGERADOR"**

Conhecido na  
computação como  
concatenação



# Expressões

## ✧ Tipos de Expressões

- Regras são essenciais para a correta avaliação de expressões
  - Operadores de mais alta prioridade devem ser avaliados primeiro. Em caso de empate, a avaliação se faz da esquerda para a direita
  - O uso de **parênteses** em sub-expressões força a avaliação das mesmas com maior prioridade





# Expressões

## ✧ Síntese

- Uma **expressão** é uma combinação de variáveis, constantes e operadores, que resulta num valor quando avaliada.
- As expressões são classificadas de acordo com o valor resultante de sua avaliação em:
  - **Aritméticas**, que resultam num valor numérico (real ou inteiro);
  - **Lógicas**, que resultam num valor lógico;
  - **Literais**, que resultam num valor literal
- Os operadores são do tipo
  - Binários
  - Unários
  - Relacionais



# Exercício

## 1. Dada a declaração de variáveis:

<b>VAR</b>	A, B, C	:	<b>inteiro</b>
	X, Y, Z	:	<b>real</b>
	NOME, RUA:		<b>literal[20]</b>
	L1, L2	:	<b>lógico</b>

Classifique as expressões seguintes de acordo com o tipo de dado do resultado de sua avaliação, em **I** (inteiro), **R** (real), **L** (literal), **B** (lógico) ou **N** (quando não for possível defini-lo):

<input type="checkbox"/> $A + B + C$	<input type="checkbox"/> $A + B + Z$	<input type="checkbox"/> $NOME + RUA$	<input type="checkbox"/> $A B$
<input type="checkbox"/> $A Y$	<input type="checkbox"/> $NOME + RUA$	<input type="checkbox"/> $L1 \text{ ou } L2$	<input type="checkbox"/> $RUA \neq NOME$
<input type="checkbox"/> $A + B / C$	<input type="checkbox"/> $A + X / Z$	<input type="checkbox"/> $A + Z / A$	<input type="checkbox"/> $A B = L1$
<input type="checkbox"/> $(A = B)$	<input type="checkbox"/> $X + Y / Z$	<input type="checkbox"/> $X = Z / A$	<input type="checkbox"/> $L1 \wedge L2$
<input type="checkbox"/> $A + B / L2$	<input type="checkbox"/> $X < L1 / RUA$		



# Instruções Primitivas

✧ ***Instruções Primitivas*** são os comandos básicos que **efetuem tarefas essenciais** para a operação dos computadores, como **entrada e saída de dados** (comunicação com o usuário e com os dispositivos periféricos), e **movimentação dos mesmos na memória**.

✧ Podem ser do tipo:

- Entrada de dados
- Saída de dados
- Atribuição

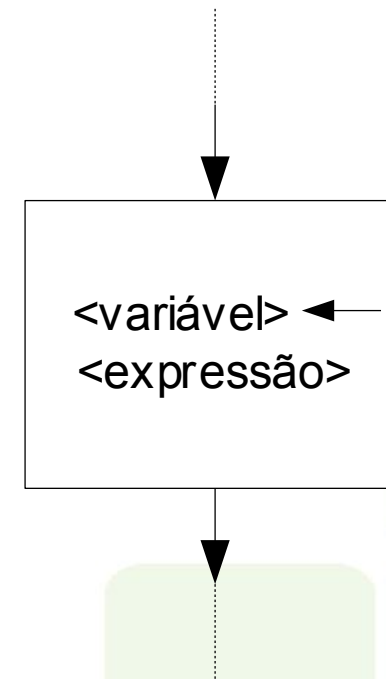


# Instruções Primitivas

## ✧ Instrução Primitiva de Atribuição

- A **instrução primitiva de atribuição**, ou simplesmente **atribuição**, é a principal maneira de se armazenar uma informação numa variável.
- Sua sintaxe é:

`<nome_de_variável> ← <expressão>`





# Instruções Primitivas

## ✧ Instrução Primitiva de Atribuição

### Pseudocódigo

**Algoritmo** EXEMPLO

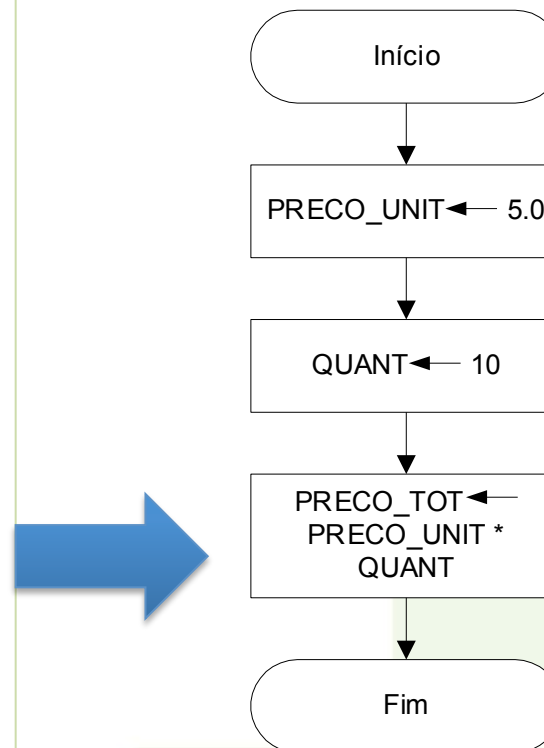
**Var** PRECO\_UNIT, PRECO\_TOT : **real**  
      QUANT : **inteiro**

**Início**

```
PRECO_UNIT ← 5.0
QUANT ← 10
PRECO_TOT ← PRECO_UNIT * QUANT
```

**Fim.**

## ✧ Fluxograma





# Instruções Primitivas

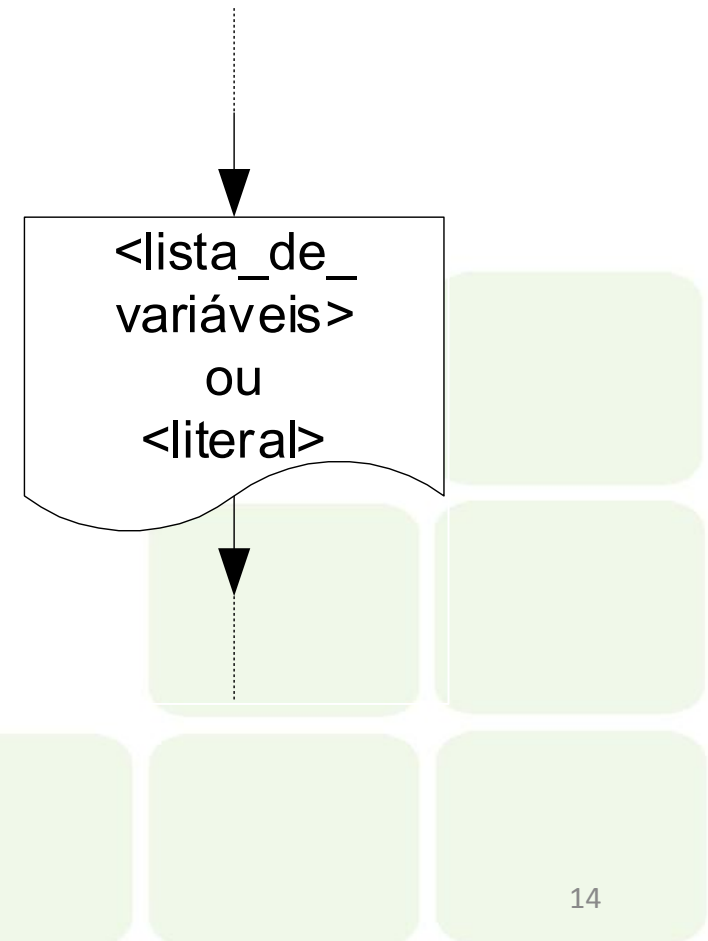
## ✧ Instrução Primitiva de Saída de Dados

- As instruções primitivas de saída de dados são o meio pelo qual informações contidas na memória dos computadores são colocadas nos dispositivos de saída, para que o usuário possa visualizá-las.
- Há duas sintaxes possíveis para esta instrução:

***Escreva <lista de variáveis>***

ou

***Escreva <literal>***





# Instruções Primitivas

## ✧ Instrução Primitiva de Saída de Dados

### Pseudocódigo

**Algoritmo** EXEMPLO

**Var** PRECO\_UNIT, PRECO\_TOT : real  
      QUANT : inteiro

**Início**

PRECO\_UNIT ← 5.0

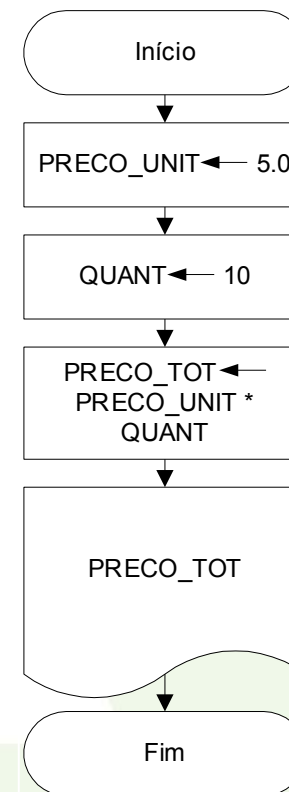
QUANT ← 10

PRECO\_TOT ← PRECO\_UNIT \* QUANT

**Escreva** (PRECO\_TOT)

**Fim.**

## ✧ Fluxograma

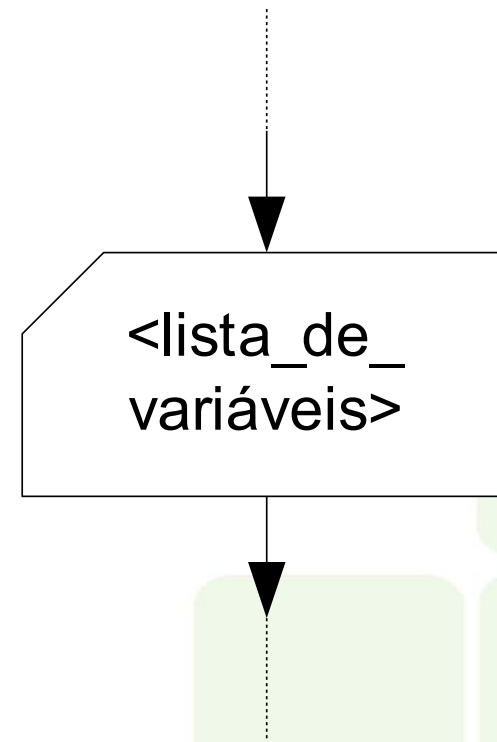




# Instruções Primitivas

## ✧ Instrução Primitiva de Entrada de Dados

- As instruções primitivas de entrada de dados são o meio pelo qual informações são fornecidas ao computador para serem processadas.
- Sua sintaxe é:  
**Leia** <lista\_de\_variáveis>







# Instruções Primitivas

## ✧ Instrução Primitiva de Entrada de Dados

### Pseudocódigo

**Algoritmo** EXEMPLO

**Var** PRECO\_UNIT,  
PRECO\_TOT : real  
QUANT : inteiro

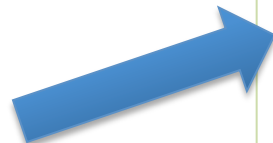
**Início**

**Leia** (PRECO\_UNIT, QUANT)

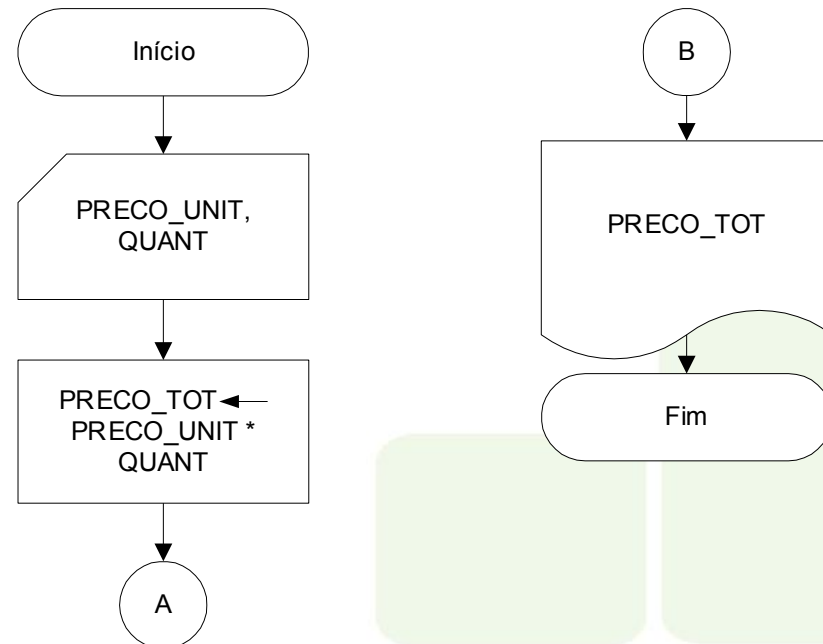
PRECO\_TOT  $\leftarrow$  PRECO\_UNIT \* QUANT

**Escreva** (PRECO\_TOT)

**Fim.**



## ✧ Fluxograma





# Instruções Primitivas

## ✧ Síntese

- A **instrução primitiva de atribuição** avalia uma expressão e armazena o valor resultante numa variável. O valor resultante da expressão e a variável devem ter tipos compatíveis.
- A **instrução primitiva de saída de dados** admite como argumentos uma lista de variáveis, um literal, ou uma mistura de ambos. No primeiro caso, o valor de cada uma das variáveis é buscado na memória e colocado no dispositivo de saída. No caso de literais, estes são copiados diretamente no dispositivo de saída.
- A **instrução primitiva de entrada de dados** busca, no dispositivo de entrada, dados que são guardados nas posições de memória correspondentes às variáveis da lista que lhe são passadas como argumento.



## Extensão da aula

- ✧ Assistir a vídeo da aula 3
- ✧ Realizar estudo sobre expressões e instruções de primitivas
- ✧ Resolver a lista de exercícios nº 1

