



ALGORITMO E LÓGICA DE PROGRAMAÇÃO

Autor: Cipriano Inácio
Área: 2025/26



REPRESENTAÇÃO EM PSEUDOCODIGO

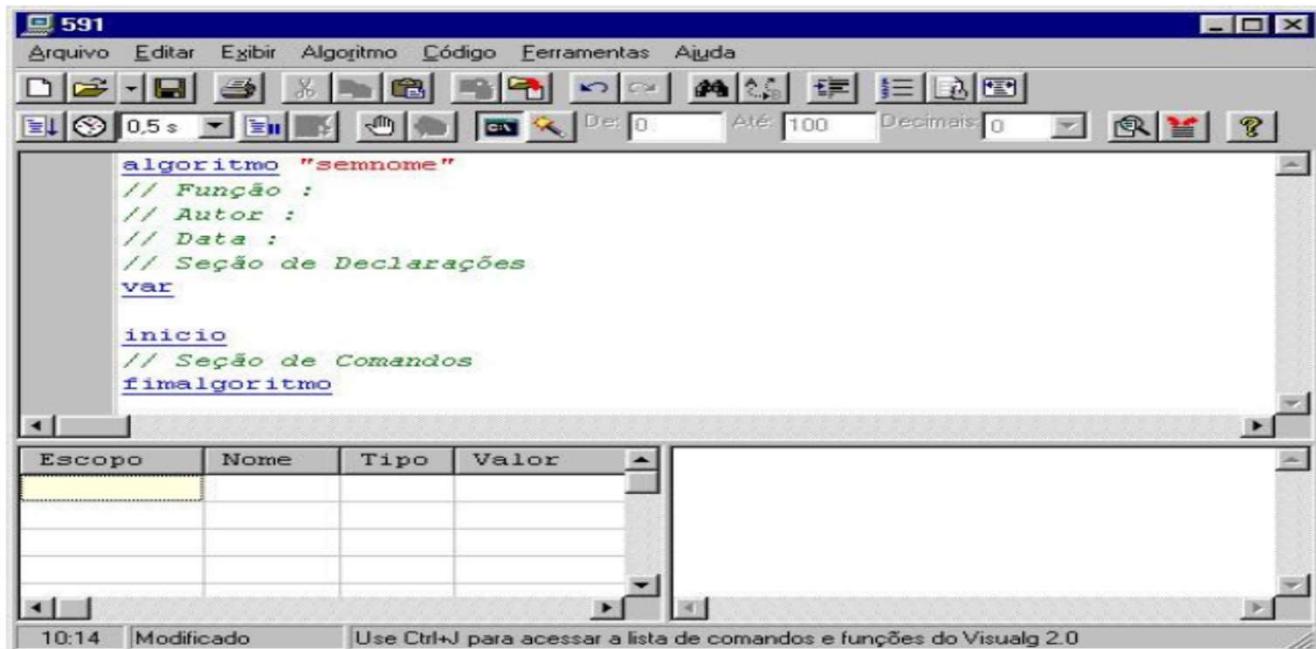


Autor: Cipriano Inácio
Ano Académico: 2025/26

- ❖ VisuAlg como ferramenta de apoio
- ❖ Compiladores e Interpretadores
- ❖ Estrutura de uma Algoritmo em VisuAlg
- ❖ Declaração de Variáveis
 - ✓ Tipos de dados primitivos
 - ✓ Variáveis e Constantes
- ❖ Expressões e Operadores
- ❖ Instrução de entrada
 - ✓ Sintaxe
- ❖ Instrução de saída
 - ✓ Sintaxe



O VisuAlg é um programa que edita, interpreta e executa algoritmos com uma linguagem próxima do português estruturado como um programa normal de Computador.



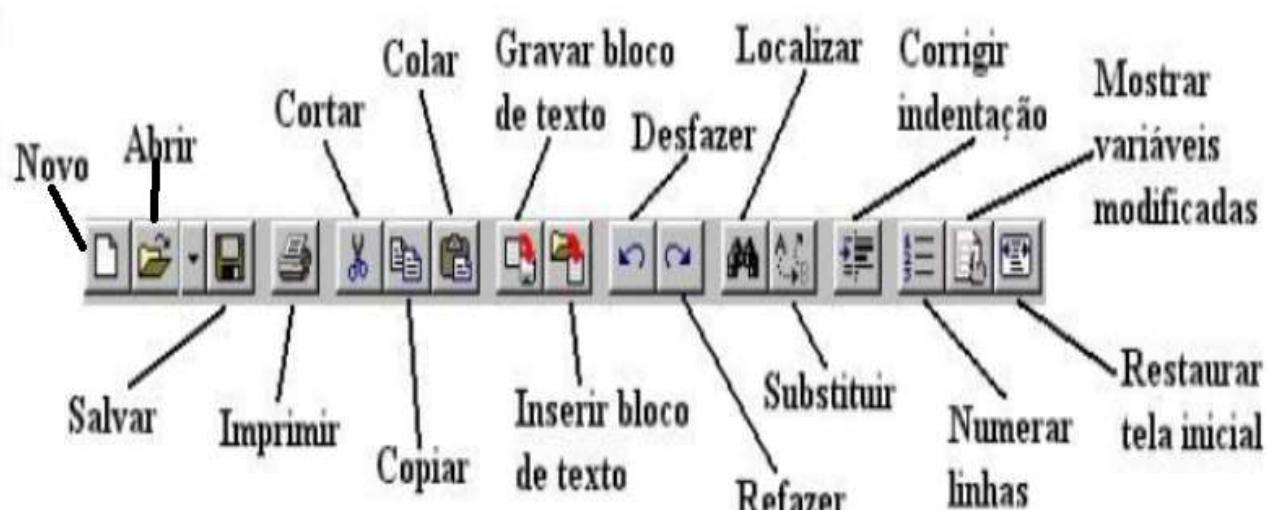
The screenshot shows the Visualg 2.0 application window. The menu bar includes Arquivo, Editar, Exibir, Algoritmo, Código, Ferramentas, and Ajuda. The toolbar contains various icons for file operations, zoom, and selection. A status bar at the bottom displays the time (10:14), a modified status (Modificado), and a keyboard shortcut (Use Ctrl+J para acessar a lista de comandos e funções do Visualg 2.0). The main area contains the following algorithm code:

```
algoritmo "semnome"
// Função :
// Autor :
// Data :
// Seção de Declarações
var

início
// Seção de Comandos
fimalgoritmo
```

BARRA DE TAREFAS DO VISUALG

A **barra de tarefas** do **Visualg** é a área localizada geralmente no **topo da janela** do programa. Ela contém **ícones e menus** que facilitam o acesso a diversas funções do ambiente de desenvolvimento do Visualg.



BARRA DE TAREFAS DO VISUALG (2)



COMPILEDORES

Os algoritmos descritos em pseudo-linguagem não podem ser executados (entendidos) por computadores. É necessário ainda uma fase de tradução do algoritmo em pseudocódigo para a linguagem que a máquina entende. Essa fase é conhecida por compilação, e é realizada pelo Compilador.

- ❖ Traduz o programa escrito em uma linguagem de programação para linguagem de máquina.
- ❖ Verifica se as sintaxes na especificação dos passos foram seguidas correctamente.
- ❖ Compiladores são relacionados directamente com a máquina para a qual será feita a tradução.
- ❖ Gera um arquivo executável do programa descrito.

INTERPRETADORES

❖ O **interpretador** traduz o código fonte linha por linha durante a execução. Cada linha é interpretada e executada imediatamente. Exemplos de linguagens que utilizam **interpretadores** são Python e Java Script. O interpretador lê o código fonte e executa as instruções conforme são encontradas.



COMPILEDORES E INTERPRETADORES

❖ A escolha entre compilador e interpretador pode depender de vários fatores, como o objetivo da linguagem, a eficiência do código gerado, a portabilidade, entre outros. Por exemplo, linguagens compiladas tendem a ser mais eficientes em termos de desempenho, pois o código já está traduzido para a linguagem da máquina. Por outro lado, linguagens interpretadas podem ser mais flexíveis e portáveis, pois não dependem de um compilador específico para cada plataforma.



- ❖ **Linguagem de Programação** é uma **linguagem** escrita e formal que especifica um conjunto de instruções e regras usadas para gerar programas (software).
- ❖ Um software pode ser desenvolvido para rodar em um computador, dispositivo móvel ou em qualquer equipamento que permita sua execução.



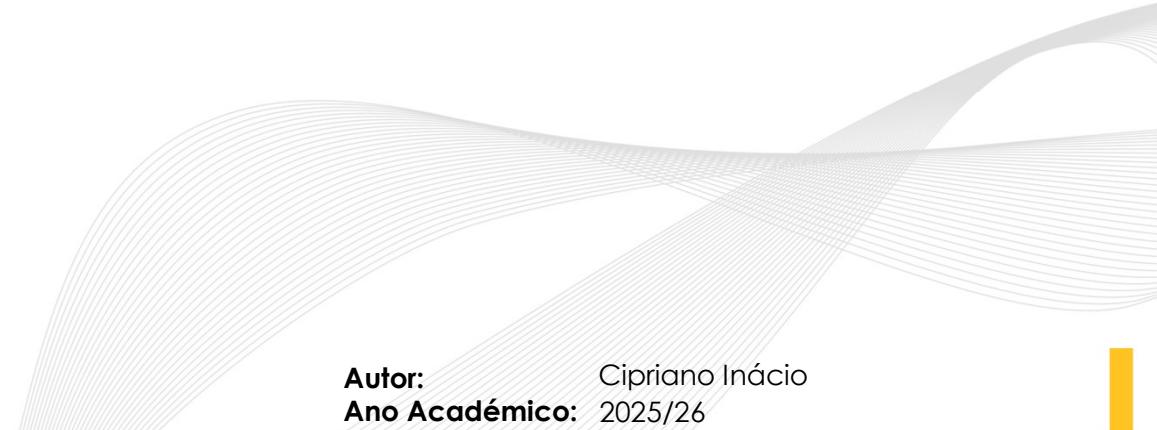
Algoritmo “ Nomealgoritmo” // Cabeçalho do Algoritmo

// Área de declaração das variáveis globais

inicio

// Corpo do algoritmo

Fimalgoritmo



Autor: Cipriano Inácio
Ano Académico: 2025/26

- ❖ Uma **variável** representa uma localização(endereço) de memória na qual podemos guardar um valor de **um dado tipo**.
- ❖ A cada **variável** está associada um **tipo de dados** e um **identificador(nome** da variável).
- ❖ O **nome de uma variável** começa obrigatoriamente por uma letra
- ❖ O **nome de uma variável** não deve conter nenhum símbolo especial excepto o underscore (_).
- ❖ O **nome**(identificador) de uma **variável** não pode ser uma **palavra reservada da linguagem**.
- ❖ Todas as **variáveis** devem ser declaradas antes de serem utilizadas.



- ❖ A maior parte das pessoas não ligadas à área de informática ignoram o potencial dos computadores e imagina que eles são capazes de tratar apenas dados numéricos.
- ❖ Definem o tipo de informação que vai ser tratada e podem ser:
 - ✓ Numéricos, Literais ou Lógicos



❖ O tipo de dado numérico pode ser:

✓ **Inteiro**: Não têm parte fracionária e podem ser positivos ou negativos;

2024 : Número inteiro positivo

0 : Número inteiro

- 8 : Número inteiro negativo

✓ **Real**: podem ter parte fracionária e podem ser positivos ou negativos.

2024.01 : Número real positivo com duas casas decimais

- 15.0 : Número real negativo com uma casa decimal

7. : Número real positivo com zero casas decimais

9.8 : Número real positivo com uma casa decimal



- ❖ O tipo de dado literal poder ser:
 - ✓ **caractere**: dado literal com apenas uma letra(cumprimento 1);
 - “a” : Caractere de comprimento 1
 - “9” : Caractere de comprimento 1
 - “+” : Caractere de comprimento 1
 - “verdadeiro” : Caractere de comprimento 10
 - “X=15.0” : Caractere de comprimento 6
 - “9.8” : Caractere de comprimento 3



- ❖ Também são chamados de **booleanos**
 - ✓ Devido a contribuição de BOOLE à área da lógica matemática.
- ❖ É identificado por **logico**
- ❖ É usado para representar dois valores únicos lógicos possíveis:
 - ✓ **verdadeiro**
 - ✓ **falso**



- ❖ **Sintaxe:** É o conjunto de regras que definem quais as relações validas entre os componentes da linguagem de modo que os comandos sejam entendidos pelo tradutor de programas.
 - ✓ Os erros de sintaxe fazem com que o nosso programa não reconheça os nossos comandos
- ❖ **Semântica:** É o significado, ou seja, o conjunto de ações que serão feitas pelo computador durante a execução do referido comando.



- ❖ As variáveis podem ser declaradas de duas formas
 - ✓ Variável: Serão aquelas que o seu valor pode variar durante a execução do algoritmo
 - Sintaxe para declaração de variáveis variavel:
 - **Var nome_da_variável: tipo_de_dado**
 - ✓ Constante: Serão aquelas que o seu valor não variam durante a execução do algoritmo



- ❖ Numa **mesma linha** poderão ser declaradas **uma ou mais variáveis do mesmo tipo**. Para tal, deve-se separar os nomes das mesmas por vírgulas.
 - ❖ Var nome_da_variavel1, nome_da_variavel2: tipo_de_dado
- ❖ Variáveis de **tipos diferentes** devem ser declaradas em **linhas diferentes**.

Var nome_da_variavel1: tipo_de_dado1

nome_da_variavel3: tipo_de_dado3



DECLARAÇÃO DE VARIÁVEIS

- ❖ Declaração de variáveis para armazenar as 3 idades, o peso, o sexo e o nome de uma pessoa.

`Var peso: real`

`idade1, idade2, idade3: inteiro`

`genero, nome: caractere`



- ❖ **Expressão** é uma combinação de **variáveis, constantes e operadores**, e que, uma vez avaliada, resulta num valor.

9+13

Area = base*altura/2

A - 8

9.8 / 2

expressão matemática

Area = $3,14159 \times \text{raio}^2$

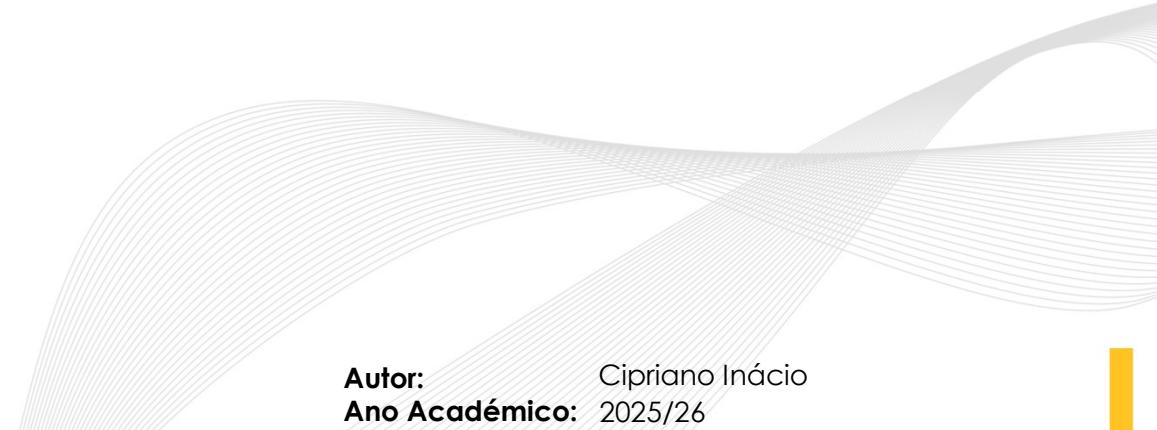
expressão computacional

Area := 3.14159*raio*raio



❖ **Operadores** : São elementos funcionais que actuam sobre **operandos** e produzem um determinado resultado.

- ✓ **Binários**: quando actuam sobre dois operandos
 - Ex.: os operadores das operações aritméticas básicas (soma, subtração, multiplicação, resto da divisão e divisão);
- ✓ **Unários**: quando actuam sobre um único operando
 - Ex.: o sinal de (-) na frente de um número, cuja função é inverter seu sinal.



OPERADORES

❖ Operadores aritméticos

OPERADOR	TIPO	OPERAÇÃO	PRIORIDADE
+	Binário	Adição	4
-	Binário	Subtração	4
*	Binário	Multiplicação	3
/	Binário	Divisão	3
\wedge	Binário	Potenciação	2
-	Unário	Inversão de sinal	3
%	Binário	Resto da divisão inteira	-

❖ Operadores de atribuição

OPERADOR	TIPO	OPERAÇÃO
$:=$ ou \leftarrow	Binário	Atribuição de valores ou resultado de uma expressão a uma variável

❖ Operadores Relacionais

OPERADOR	TIPO	OPERAÇÃO
=	Binário	Igual
<>	Binário	Diferente
<=	Binário	Menor ou igual
>=	Binário	Maior ou igual
>	Binário	Maior
<	Binário	Menor

❖ Operadores Lógicos

OPERADOR	TIPO	OPERAÇÃO
ou	Binário	Disjunção
e	Binário	Conjunção
nao	Unário	Negação

- ❖ **Dispositivo de entrada** é o meio pelo qual as **informações** (mais especificamente os **dados**) são enviadas pelo utilizador ao computador.
 - ✓ Teclado, Mouse, Scanner, Lápis Optico, Scanner, Modem, Unidade CD

- ❖ **Dispositivo de saída** é o meio pelo qual as informações (geralmente, os resultados da execução de um programa) são disponibilizadas pelo computador ao utilizador
 - ✓ Monitor de vídeo, Impressora, Auscultadores, Colunas;



- ❖ A instrução primitiva de entrada de dados permite a inserção de dados pra o computador usando o teclado
 - ✓ Sintaxe:
 - ✓ Leia(Variável ou Lista de Variáveis)



Exemplo:

Var nome: caratere // Declaração da variável nome

Leia(nome) // Entrada ou atribuição do valor do nome pelo teclado



- ❖ Permitem que as informações contidas na memória dos computadores sejam colocados nos dispositivos de saída de modo que o utilizador as possa visualizar
 - ✓ Sintaxe:
 - ✓ escreva(Constantes, Expressões ou Lista de Variáveis)

Exemplo:

`Var Idade: inteiro // Declaração da variável Idade`



`escreva("Valor da Idade: ",idade,8+5) // Mostra na tela o seguinte:`

Valor da Idade: ???? 13

Obs: ???? // Faz referência ao valor da variável Idade.



Os algoritmos deverão ser representados em Fluxograma convencional e em pseudocódigo segundo a sintaxe do portugol studio.

1. Escreva um algoritmo para calcular a média entre dois números quaisquer.
2. Escreva um algoritmo que efetue o cálculo da área de uma circunferência, apresentando a área calculada.
3. Escreva um algoritmo que calcule a área e o perímetro de um triângulo
4. Escreva um algoritmo que calcule o quadrado de um número introduzido pelo utilizador





Volenti Nihil Difficili - “A quem quer, nada é difícil”