

Roteiro:

- Conceitos básicos de algoritmo, linguagem, processador de linguagem e ambiente de programação;
- Aspectos fundamentais da organização e do funcionamento de um computador;
- Construções básicas de linguagens e execução sequencial;
- Ambiente Visualg.

Algoritmo

- Linguagem natural:
 - Ambígua
 - Imprecisa
 - Incompleta
- Pseudocódigo:
 - Portugol (livro texto)
 - Visualg (linguagem)
- Fluxograma

Algoritmo

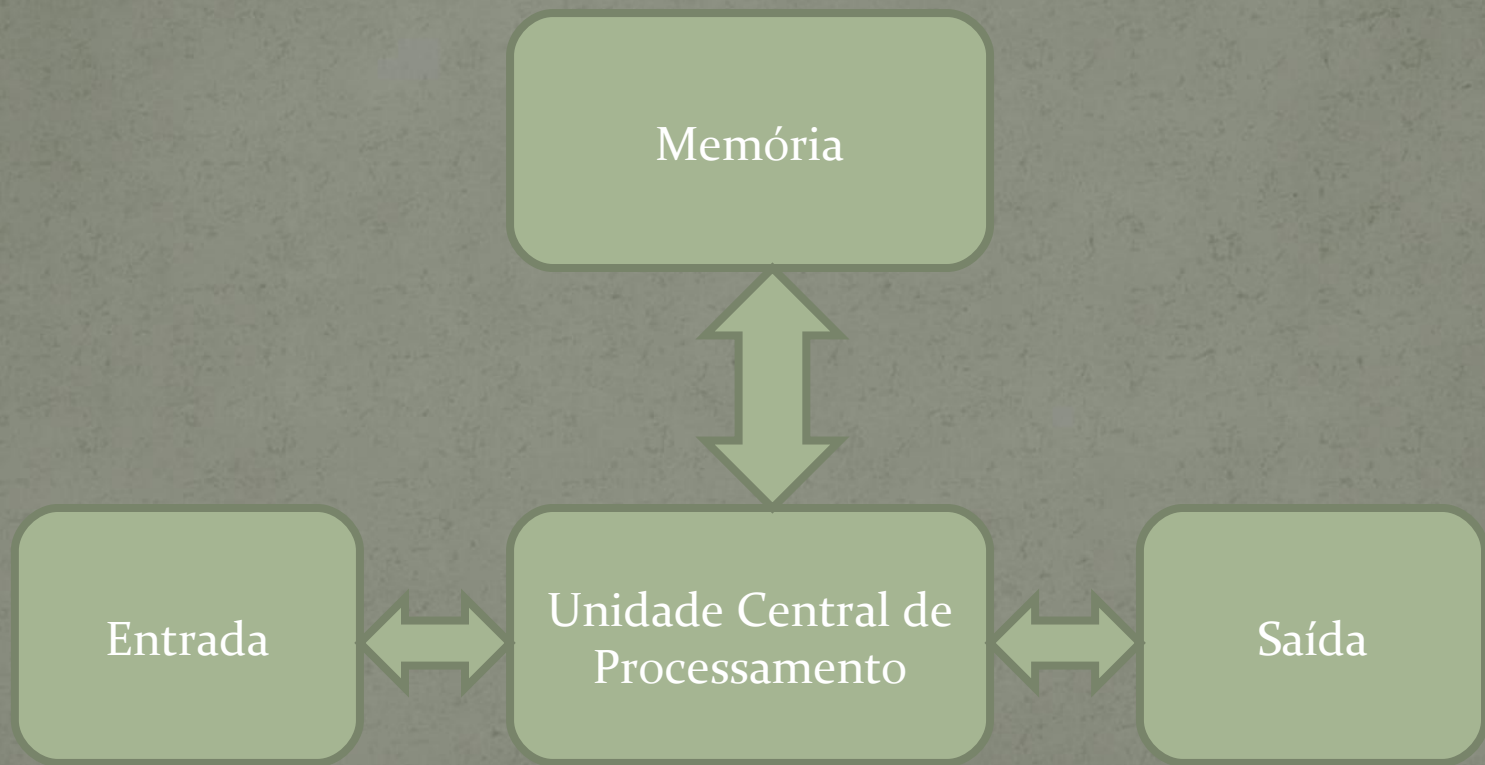
- Objetivo:

- Criar um texto (ou diagrama) que represente de forma clara os passos que devem ser executados para se chegar na solução.
- Permitir que a tradução para uma linguagem de programação seja feita de forma direta e sem grandes dificuldades.
- Eventualmente, o próprio computador poderá ser usado para se escrever e testar o algoritmo concebido.

Algoritmo

- Visualg:
 - Uma linguagem para representação de algoritmos;
 - Um processador que permite escrever, armazenar, recuperar e depurar algoritmos;
 - Ligeiramente diferente da linguagem utilizada pelo livro-texto (Portugol);

Modelo de von Neumann



Modelo de von Neumann

- Os dados e os programas são armazenados na memória, em regiões distintas.
- Os programas são formados, essencialmente, por comandos (instruções sobre o que fazer);
- Os comandos são lidos sequencialmente da memória, um após o outro;
- A execução de um novo comando inicia apenas depois que a execução do anterior tiver terminado (execução sequencial).
- Eventualmente, um comando pode modificar o valor de um dado existente na memória, solicitar novos dados ao usuário ou enviar dados para a saída.

Programa
(e Algoritmo)

=

Dados + Instruções

Variáveis

- “Variável” é o nome que se dá para uma região da memória onde é armazenado algum dado que está ou estará sendo manipulado pelo programa.
- Um programa pode manipular várias variáveis distintas;
- Cada variável pode armazenar vários valores, mas apenas um de cada vez;
- “Variáveis” são criadas no início da execução do programa e destruídas ao término da sua execução;
- Uma vez atribuído um valor para uma variável, este permanece até que sofra nova modificação pelo programa;
- O conjunto de “variáveis” que um programa necessita precisa ser definido antes de se iniciar a execução do programa;
- As “variáveis” são referenciadas por intermédio de nomes (identificadores), que devem ser distintos entre si.

Tipos

- Toda “variável” precisa estar associada a algum “tipo” de dados
- O “tipo” de uma variável determina a coleção finita de valores que podem ser atribuídos à mesma;
- O “tipo” de uma variável é fixo durante toda a execução do programa.
- Os “tipos” de todas as “variáveis” precisam ser definidos antes de se iniciar a execução do programa.

Declarações

- Seqüência de instruções que servem para informar quais variáveis estarão sendo usadas pelo programa e quais os seus respectivos tipos;
- Não é possível mudar o tipo de uma variável durante a execução do programa;
- Não é possível criar ou destruir variáveis durante a execução do programa;
- Tudo precisa ser planejado antes – durante a elaboração do algoritmo.

Comandos

- Determinam quando e quais ações “primitivas” devem ser executadas;
- São exemplos de ações “primitivas”: leitura de dados, saída de dados, atribuição de valor a uma variável;
- Além disso, os comandos podem ser “estruturados”;
- A “estruturação” dos comandos permite que eles sejam executados numa determinada ordem, que a sua execução seja repetida ou que se opte pela escolha de um ou outro comando subordinado.
- Basicamente, a “estruturação” dos comandos permite o estabelecimento de um “fluxo de controle”, ou seja, uma seqüência de execução de ações primitivas através do qual se pretende alcançar a solução do problema original.

Portanto, algoritmos e
programas...

são compostos por...

Declarações e Comandos.

A linguagem Visualg

Estrutura geral

```
algoritmo "semnome"  
// Seção de Declarações  
inicio  
// Seção de Comandos  
fimalgoritmo
```

Sublinhado: palavras-chave da linguagem. Possuem significado pré-definido.
// Indica uma linha de comentário. Ignorada pelo processador.
Uma declaração ou comando em cada linha.

A linguagem Visualg

Declaração de variáveis

var <lista-de-variáveis> : <tipo-de-dado>

<lista-de-variáveis>

Seqüência de um ou mais identificadores separados por vírgula.

<identificador>

Seqüência de até 30 caracteres. O primeiro deve ser uma letra e os demais letras, dígitos ou o caracter “_”; Maiúsculas e minúsculas não são diferenciadas.

A linguagem Visualg

Tipos de dados

<tipo-de-dado>

- inteiro: define variáveis numéricas do tipo inteiro, ou seja, sem casas decimais.
- real: define variáveis numéricas do tipo real, ou seja, com casas decimais.
- caractere: define variáveis do tipo *string*, ou seja, cadeia de caracteres.
- logico: define variáveis do tipo *booleano*, ou seja, com valor VERDADEIRO ou FALSO.

A linguagem Visualg

Exemplo

var a: inteiro

Valor1, Valor2: real

vet: vetor [1..10] de real

matriz: vetor [0..4,8..10] de inteiro

nome_do_aluno: caractere

senalizador: logico

A linguagem Visualg

Comando de atribuição

<identificador> <- <expressão>

Valor calculado é armazenado na variável *<identificador>*. O valor anterior é descartado.

<expressão>

- Fórmula matemática que, após avaliada, resulta num valor.
- Podem ser usados os operadores mais comuns (soma, subtração, divisão, multiplicação etc).
- O tipo do valor resultante deve ser compatível com o tipo da variável denotada pelo identificador.

A linguagem Visualg

Exemplos

```
a <- 3
Valor1 <- 1.5
Valor2 <- Valor1 + a
vet[1] <- vet[1] + (a * 3)
matriz[3,9] <- a/4 - 5
nome_do_aluno <- "José da Silva"
sinalizador <- FALSO
```

A linguagem Visualg

Expressões

<expressão>

- ❖ Operandos:
 - Literais: 0, 1, 2, 3, ..., 0.15, 3.1415, ..., VERDADEIRO ou FALSO, “cadeia de caracteres”
 - Identificadores: Variáveis
- ❖ Operadores:
 - Aritméticos: +, -, *, /, \, ^, %
 - Lógicos: nao, e, ou, xou
 - Relacionais: =, <, >, <>, >=, <=
- ❖ Precedência e associatividade

A linguagem Visualg

Exemplos

```
a <- 3+b
```

```
Valor1 <- 1.5*471-23,99
```

```
Valor2 <- Valor1 ^a*b
```

```
nome_do_aluno <- "José"
```

```
nome_do_aluno <- nome_do_aluno+" da Silva"
```

```
senalizador <- FALSO
```

```
senalizador<-senalizador e VERDADEIRO
```

A linguagem Visualg

Saída de dados

escreva (<lista-de-expressões>)

escreval (<lista-de-expressões>)

- ❖ As expressões são avaliadas, uma a uma, da esquerda para a direita, e o correspondente valor é enviado para o dispositivo de saída padrão.

```
algoritmo "exemplo"
```

```
var x: real
```

```
    y: inteiro
```

```
    a: caractere
```

```
    l: logico
```

```
inicio
```

```
x <- 2.5
```

```
y <- 6
```

```
a <- "teste"
```

```
l <- VERDADEIRO
```

```
escreval ("x", x:4:1, y+3:4) // Escreve: x 2.5 9
```

```
escreval (a, "ok") // Escreve: testeok (e depois pula linha)
```

```
escreval (a, " ok") // Escreve: teste ok (e depois pula linha)
```

```
escreval (a + " ok") // Escreve: teste ok (e depois pula linha)
```

```
escreva (l) // Escreve: VERDADEIRO
```

```
fimalgoritmo
```

A linguagem Visualg

Entrada de dados

leia (<lista-de-variáveis>)

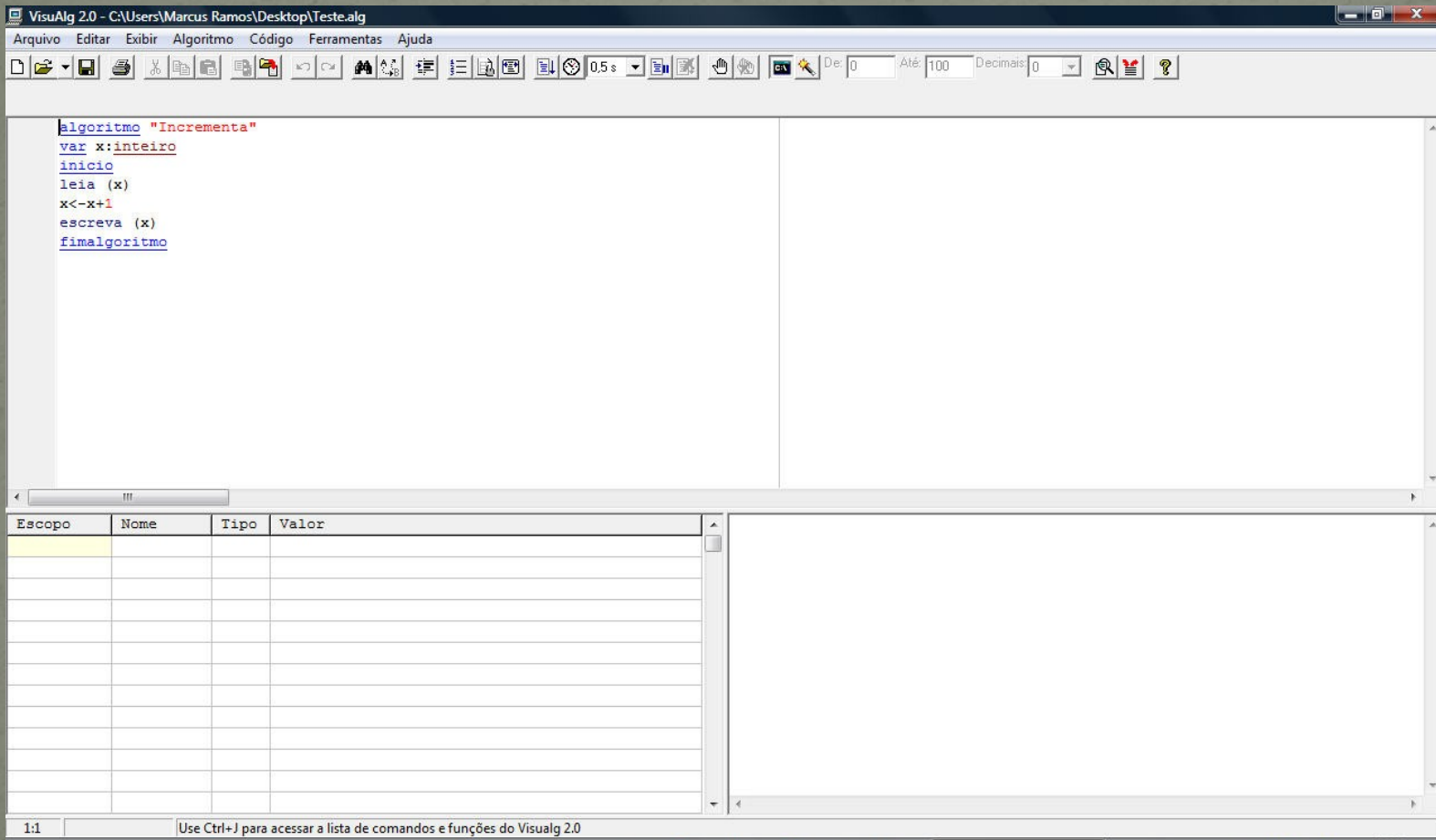
- ❖ São feitas diversas leituras de dados, uma para cada variável presente na lista. Após a leitura, o valor é armazenado na variável correspondente.

A linguagem Visualg

Exemplo

```
algoritmo "Incrementa"  
var x: inteiro;  
inicio  
leia (x)  
x<-x+1  
escreva (x)  
fimalgoritmo
```


O ambiente Visualg



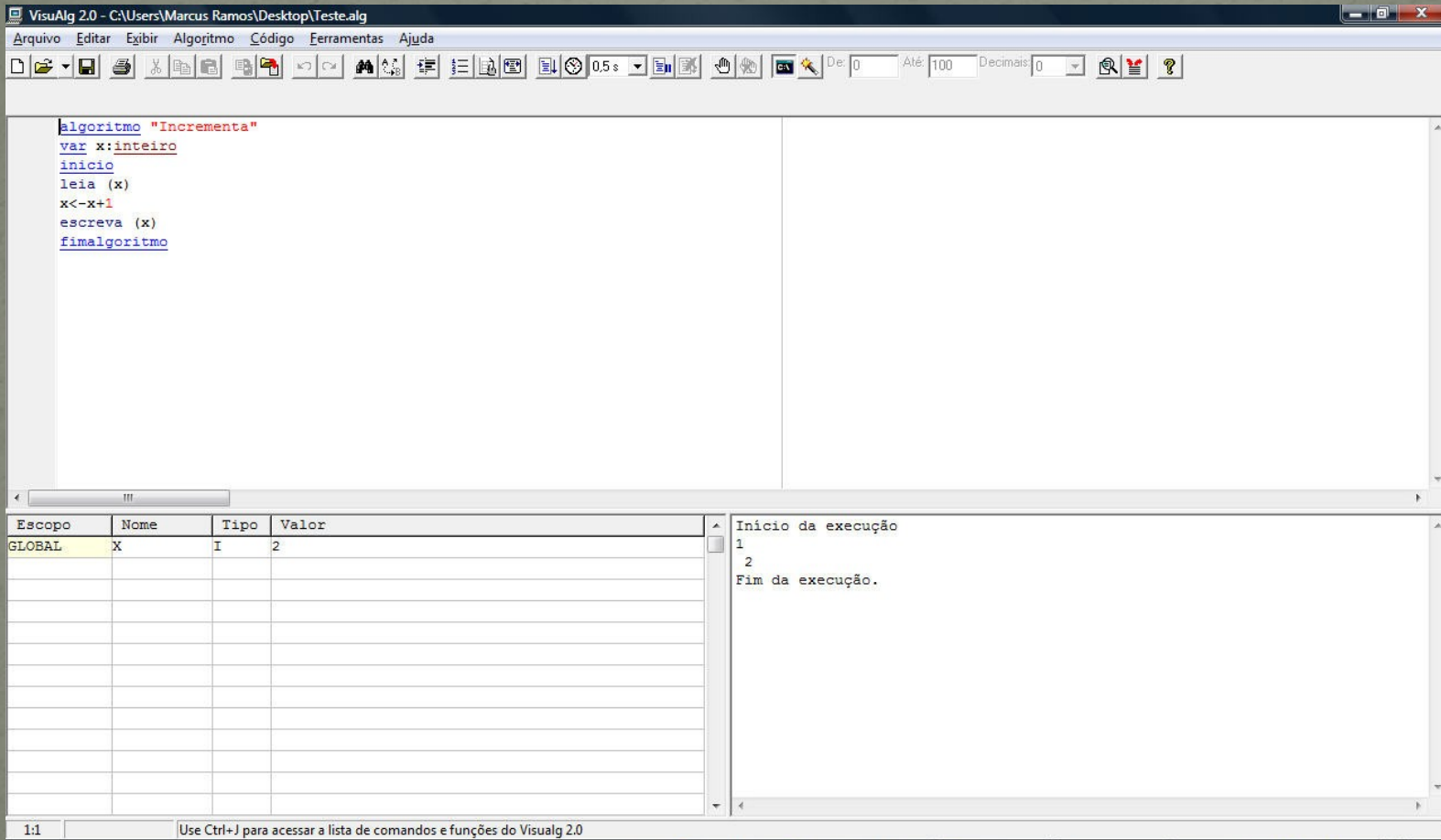
O ambiente Visualg

algoritmo "Incrementa"
var x:inteiro
inicio
leia (x)
x<-x+1
escreva (x)
fimalgoritmo

Escopo	Nome	Tipo	Valor
GLOBAL	X	I	0

1:1 Use Ctrl+J para acessar a lista de comandos e funções do Visualg 2.0

O ambiente Visualg



The screenshot displays the Visualg 2.0 application window. The title bar shows the file path: C:\Users\Marcus Ramos\Desktop\Teste.alg. The menu bar includes Arquivo, Editar, Exibir, Algoritmo, Código, Ferramentas, and Ajuda. The toolbar contains various icons for file operations, editing, and execution. The code editor contains the following pseudocode:

```
algoritmo "Incrementa"  
var x:inteiro  
inicio  
leia (x)  
x<-x+1  
escreva (x)  
fimalgoritmo
```

Below the code editor is a table with the following data:

Escopo	Nome	Tipo	Valor
GLOBAL	X	I	2

To the right of the table is a console window showing the execution output:

```
Início da execução  
1  
2  
Fim da execução.
```

At the bottom of the window, a status bar indicates: 1:1 Use Ctrl+J para acessar a lista de comandos e funções do Visualg 2.0.