

## 2ª Aula

# Tema : Instruções e programas: pseudo-linguagem

Prof. Cícero Roberto Ferreira de Almeida

28 de agosto de 2014

## Sumário

Instruções e programas: pseudo-linguagem  
Operadores e Expressões  
Expressões lógicas  
Comando de atribuição  
Comandos de entrada e saída

- 1 Instruções e programas: pseudo-linguagem
- 2 Operadores e Expressões
- 3 Expressões lógicas
- 4 Comando de atribuição
- 5 Comandos de entrada e saída

## Sumário

Instruções e programas: pseudo-linguagem  
Operadores e Expressões  
Expressões lógicas  
Comando de atribuição  
Comandos de entrada e saída

*“A forma pela qual você olha para um problema determina se você o encarra ou corre dele. Tente olhá-lo sempre de igual para igual, sem menosprezar, sem temer”*

## Pseudocódigo - *pseudo-linguagem*

- Os algoritmos são construídos utilizando uma linguagem chamada pseudocódigo;
- Livre de detalhes de computação;
- Não possuem um formalismo rígido como as linguagens de programação, mas convenções;
- Dar-se o nome de **Portugol** à pseudo-linguagem utilizada na construção de algoritmos ou também, **Português Estruturado**.

Sumário
Instruções e programas: pseudo-linguagem
Operadores e Expressões
Expressões lógicas
Comando de atribuição
Comandos de entrada e saída

Portugol
Regras para construção de Algoritmos
Estrutura de um Algoritmo
Visualg
Tipos de dados
Variável
Constante
Formação de Identificadores
Declaração de variáveis
Linearização de Expressões

# Portugol

É uma pseudolinguagem de programação (simbiose do Português com o ALGOL e PASCAL). O Portugol fornece uma notação para algoritmos, a ser utilizada na definição, na criação, no desenvolvimento e na documentação de um programa.

## Regras para construção de Algoritmos

- Usar somente uma ação por frase;
- Imaginar que você está desenvolvendo um algoritmo para pessoas não técnicas;
- Usar sentenças curtas e simples;
- Especificar ações claras e precisas, que a partir de um estado inicial, após um período de tempo finito, produzem um estado final previsível e bem definido;
- Garantir que sempre que executado, sob as mesmas circunstâncias produza o mesmo resultado.

## Regras para construção de Algoritmos

- Um algoritmo inicia-se com a palavra reservada *Algoritmo* seguida de um nome, da declaração de variáveis e do bloco principal;
- Um bloco pode ser definido como um conjunto de comandos com uma função bem definida.

# Estrutura de um Algoritmo

**Algoritmo** <nome do algoritmo>

<declaração de variáveis>

**inicio**

<comandos>

**fim do algoritmo**



# Visualg

- Será utilizado nessa disciplina o Visualg;
- O Visualg é um programa que edita, interpreta e executa algoritmos como um programa normal de computador;
- Utiliza o Portugol;
- Gratuito, disponível para download no endereço;  
*<http://www.guanabara.info/logica/Ferramentas/visualgv25.exe>*
- Favorece o aprendizado, uma vez que o aluno tem a opção de visualizar a execução dos seus algoritmos, sem no entanto, conhecer uma linguagem de programação;
- **O Visualg não é uma linguagem de programação**

# Tipos de dados

- Os tipos básicos utilizados na construção de algoritmos, também conhecidos como **Tipos Primitivos**, são:
  - Inteiro**  $\Rightarrow$  toda e qualquer informação prévia numérica que pertença ao conjunto dos números inteiros relativos (negativa, nula ou positiva). Exemplo:
    - Ele tem 16 anos;
    - A escada possui 8 degraus;
    - A sala possui 40 cadeiras.
  - Real**  $\Rightarrow$  toda e qualquer informação numérica que pertença ao conjunto dos números reais (negativa, nula ou positiva). Exemplo:
    - Ela tem 1,73 metro de altura;
    - O saldo bancário de José é de R\$ 200,30.

# Tipos de dados

- **Caractere**  $\Rightarrow$  toda e qualquer informação composta de um conjunto de caracteres alfanuméricos: numéricos (0..9), alfabéticos (a..z, A..Z) e os caracteres especiais. Exemplo:
  - 1 O nome do vencedor é Isaque Sampaio;
  - 2 O parque está repleto de placas: “Não pise na grama”.
- **Lógico**  $\Rightarrow$  toda e qualquer informação que pode assumir apenas duas situações. Exemplo:
  - 1 A porta pode estar aberta ou fechada;
  - 2 A lâmpada pode estar acesa ou apagada.

# Variável

Variável é um elemento através do qual armazena-se durante a execução do algoritmo um determinado valor, podendo alterá-lo a qualquer instante durante a execução do algoritmo.

- Exemplo
  - 1 Cotação do dólar;
  - 2 Peso de uma pessoa;
  - 3 Índice da inflação;

# Constante

É um tipo especial de variável, a qual não terá seu conteúdo modificado durante toda a execução do algoritmo, ou seja, seu valor é fixo e determinado no instante de construção do algoritmo.

- Exemplo

①  $\pi \Rightarrow 3,14159265$

## Formação de Identificadores

- A escolha do nome das variáveis e constantes de um algoritmo segue uma regra que dever ser seguida;
- Esses nomes são chamados de **identificadores**, os quais devem obedecer as seguintes regras de formação:
  - 1 Devem iniciar com uma letra do alfabeto;
  - 2 Podem ser seguidos por mais caracteres alfabéticos, numéricos ou sinal de sublinhar (underscore);
  - 3 Não é permitido que o identificador comece com um número, **ou** **contenha espaços em branco** ou símbolos especiais (com exceção do underscore)



## Exemplo de Identificadores

- Válidos
  - Alpha,Media,Notas, FGTS.
- Inválidos
  - 5X, XB(432), *Notas/2*, X\*, P&AAA;



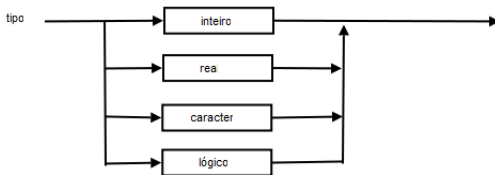
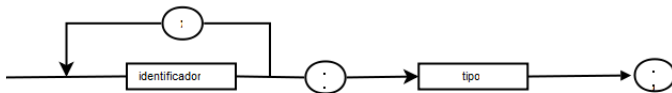
## Declaração de variáveis

- As variáveis só podem armazenar valores de um mesmo tipo, de maneira que também são classificadas como sendo numéricas, lógicas e literais.
- Para trabalhar com as variáveis é necessário declará-las segundo a regra de formação de identificadores;
- Toda variável deve ter um tipo, definido durante a construção do algoritmo;
- O tipo da variável não muda durante a execução do algoritmo.

Sumário  
Instruções e programas: pseudo-linguagem  
Operadores e Expressões  
Expressões lógicas  
Comando de atribuição  
Comandos de entrada e saída

Portugol  
Regras para construção de Algoritmos  
Estrutura de um Algoritmo  
Visualg  
Tipos de dados  
Variável  
Constante  
Formação de Identificadores  
**Declaração de variáveis**  
Linearização de Expressões

## Declaração de variáveis



## Exemplo de declaração de variáveis

- x : inteiro;
- nome : caracter;
- total,media : real;

## Linearização de Expressões

- Para a construção de algoritmos que realizam cálculo matemáticos, todas as expressões aritméticas devem ser colocadas em linhas (linearizadas), devendo ser feito o mapeamento dos operadores da aritmética tradicional para os do Português Estruturado. Exemplo:

$$\frac{2x}{4} \Rightarrow 2 * x / 4 \quad (1)$$

$$\frac{\sqrt{2x}}{y} \Rightarrow \text{sqrt}(2 * x) / y \quad (2)$$

# Operadores

- ① Operadores são utilizados para relacionar variáveis ou constantes entre si.
- ② Há três tipos de operadores:
  - Aritméticos;
  - Relacionais;
  - Lógicos;

# Operadores Aritméticos

- soma  $\Rightarrow +$
- subtração  $\Rightarrow -$
- multiplicação  $\Rightarrow *$
- divisão  $\Rightarrow /$
- Exemplo:  $5 * 2/3 = 3,33333$

## Expressões lógicas

Denominas expressões lógicas aquela cujo operadores são lógicos ou relacionais e cujo operandos são relações ou variáveis ou constantes do tipo lógico.

- Relações - é uma comparação realizada entre valores do mesmo tipo. Representados na relação por constantes, variáveis ou expressões do tipo correspondente.

Os operadores relacionais são:

- $= \Rightarrow$  igual;
- $<> \Rightarrow$  diferente;
- $<= \Rightarrow$  menor ou igual;
- $< \Rightarrow$  menor
- $> \Rightarrow$  maior
- $>= \Rightarrow$  maior ou igual
- $\in \Rightarrow$  contido em conjunto



O resultado de uma relação é sempre um valor lógico, isto é, **falso** ou **verdadeiro**.

Exemplo:

- $A <> B$ ;
- $A > B$ ;
- $A \geq B$ ;

## Operadores lógicos

A álgebra das proposições define três conectivos usados na formação de novas proposições a partir de outras conhecidas. Os operadores lógicos são:

- **NÃO**  $\Rightarrow$  para a negação
- **E**  $\Rightarrow$  para conjunção
- **OU**  $\Rightarrow$  para disjunção

Exemplo:

- Se chover **E** relampejar, eu fico em casa.
- Se chover **OU** relampejar, eu fico em casa.

## Operação de negação

Tabela verdade

<b>A</b>	<b>não A</b>
F	V
V	F

## Operação de conjunção

Tabela verdade

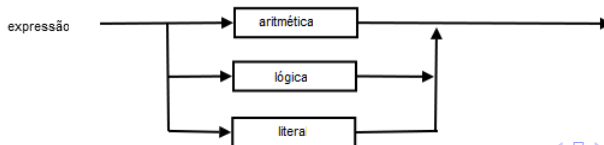
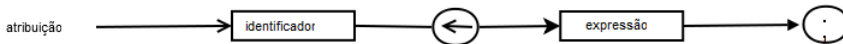
<b>A</b>	<b>B</b>	<b>A e B</b>
F	F	F
F	V	F
V	F	F
V	V	V

## Operação de disjunção não-exclusiva

Tabela verdade

A	B	A ou B
F	F	F
F	V	V
V	F	V
V	V	V

- Um comando de atribuição permite-nos fornecer um valor a uma variável;
- O comando de atribuição segue a seguinte sintaxe:



- Para que os algoritmos possa apresentar os dados que calculou como resposta ao problema que resolveu, adotamos o comando de saída de dados denominado **escreva**, cuja finalidade é exibir o conteúdo da variável identificada;
- O comando **escreva** segue a seguinte regra sintática:

