



FG  
Faculdade dos Guararapes  
LAUREATE INTERNATIONAL UNIVERSITIES



LAUREATE  
INTERNATIONAL  
UNIVERSITIES

# Algoritmo e Programação

## Capítulo 2

Tópicos Preliminares

# Tópicos Preliminares

- Neste capítulo
  - Tipos Primitivos
  - Variáveis
  - Expressões Aritméticas, Lógicas e Relacionais
  - Comandos de Entrada e Saída
  - Blocos

# Tópicos Preliminares

- Tipos Primitivos
  - São os tipos básicos de informação dos algoritmos
    - **Inteiro**: informação pertencente ao conjunto dos números inteiros relativos (positiva ou negativa)
    - **Real**: informação pertencente ao conjunto dos números reais
    - **Caracter**: informação alfanumérica (caracteres alfabéticos, numéricos e especiais)
    - **Lógico**: informação biestável (admite apenas 2 valores)

# Tópicos Preliminares

- Constantes
  - São valores que não sofrem nenhuma variação no decorrer do tempo
  - São exemplos de constantes: o valor de  $\pi$ , a velocidade da luz, 5, “Não fume”

# Variáveis

- São valores que podem sofrer alteração no decorrer do tempo. Ex: Cotação do dólar, o peso de uma pessoa, o preço da gasolina
- São como gavetas que podem receber diversos tipos de objetos
  - Identificadores: São os nomes escolhidos para as informações variáveis
    - Deve iniciar por caracter alfabético
    - Pode ser seguido por mais caracteres alfabéticos ou numéricos
    - Não devem ser usados caracteres especiais
  - Declaração: processo de reservar e etiquetar gavetas

# Expressões

- Expressões Aritméticas
  - Operadores aritméticos: utilizados para a realização de cálculos matemáticos

Operador	Função	Exemplos
<b>+</b>	<b>Adição</b>	<b>2 + 3, X + Y</b>
<b>-</b>	<b>Subtração</b>	<b>4 - 2, N - M</b>
<b>*</b>	<b>Multiplicação</b>	<b>3 * 4, A * B</b>
<b>/</b>	<b>Divisão</b>	<b>10 / 2, C / D</b>
<b>pot(x,y)</b>	<b>Potenciação (x elevado a y)</b>	<b>pot(2, 3)</b>
<b>rad(x)</b>	<b>Raiz quadrada (de x)</b>	<b>rad(9)</b>
<b>Mod</b>	<b>Resto da divisão</b>	<b>9 mod 4 resulta 1</b>
<b>Div</b>	<b>Quociente da divisão inteira</b>	<b>9 div 4 resulta 2</b>

# Expressões

- Expressões Lógicas
  - Operadores relacionais: utilizados para estabelecer relação de comparação entre valores

Operador	Função	Exemplos
=	Igual a	$3 = 3$ , $X = Y$
>	Maior que	$5 > 4$ , $X > Y$
<	Menor que	$3 < 6$ , $X < Y$
>=	Maior ou igual a	$5 >= 3$ , $X >= Y$
<=	Menor ou igual a	$3 <= 5$ , $X <= Y$
<>	Diferente de	$8 <> 9$ , $X <> Y$

# Expressões

- Expressões Lógicas
  - Operadores lógicos: utilizados para a efetuar avaliações lógicas entre valores

Operador	Função	Exemplos
Não	Negação	não V, não X
e	Conjugação	V e V, X e Y
ou	Disjunção	V ou V, X ou Y

- Tabelas Verdade: Conjunto de todas as possibilidades de cada operador lógico

A	não A
F	V
V	F

A	B	A e B
F	F	F
F	V	F
V	F	F
V	V	V

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



# Atribuição

- Processo de associar um valor a uma variável (guardar um objeto na gaveta)
  - O tipo de dado deve ser compatível com a variável (objeto precisa caber na gaveta)
  - Cada variável pode receber apenas um valor. O segundo valor sobrepõe-se ao anterior
  - Podem ser atribuídos:
    - Constantes
    - Variáveis
    - Expressões (aritméticas, relacionais ou lógicas)
  - Comando de atribuição : ←
  - Ex.:  $B \leftarrow 7$ ;  $A \leftarrow B$ ;  $X \leftarrow B + 13 \text{ div } 5$ ;

# Entrada e Saída

- Algoritmos objetivam transformar informações
- Algoritmo = Entrada + Processamento + Saída
- Entrada: obtenção de dados provenientes do meio externo
  - Comando: **leia**
  - Exemplos:
    - **leia** (X);
    - **leia** (A, NOTA);
- Saída: entrega dos resultados ao meio externo
  - Comando: **escreva**
  - Exemplos:
    - **escreva** (X);
    - **escreva** (B, MEDIA, 2+2);

# Blocos

- Um Bloco é um conjunto de ações com uma função definida
- O algoritmo pode ser visto como um Bloco
- O algoritmo pode conter vários Blocos
- Exemplo:

**início** // início do bloco (algoritmo)

*// declaração de variáveis*

*// seqüência de ações (eventualmente mais blocos)*

**fim.** // fim do bloco (algoritmo)

# Representação de Algoritmo

- **Método de representação de algoritmos** → Utilizaremos pseudocódigo.
- **Pseudocódigo** → Técnica textual de representação de um algoritmo - Também conhecida como Português Estruturado ou **Portugol**.
- Técnica é baseada em uma PDL (*Program Design Language*), que é uma linguagem genérica na qual é possível representar um algoritmo de forma semelhante à das linguagens de programação.

# Pseudocódigo

```
algoritmo "Habilitacao"  
var  
    idade: numerico  
inicio  
    escreva ("informe idade:")  
    leia(idade)  
    se idade >= 18 entao  
        escreva("pode tirar a carteira")  
    senao  
        escreva("não pode tirar a carteira")  
    fimse  
fimalgoritmo
```

# Passos para criação de um algoritmo

1. Ler atentamente o enunciado do problema, compreendendo-o e destacando os pontos mais importantes;
2. definir os dados de entrada, ou seja, quais dados serão fornecidos;
3. definir os dados de saída, ou seja, quais dados serão gerados depois do processamento;
4. definir o processamento, ou seja, quais cálculos serão efetuados e quais as restrições para esses cálculos. O processamento é responsável pela obtenção dos dados de saída com base nos dados de entrada;
5. definir as variáveis necessárias para armazenar as entradas e efetuar o processamento;
6. elaborar o algoritmo;

# Exercício

1. Escreva um algoritmo que receba dois números e calcule a média deles, exibindo uma mensagem que diga a média é: com o valor da média calculada.
2. *Faça um algoritmo que leia dois números e identifique se são iguais ou diferentes. Caso eles sejam iguais imprima uma mensagem dizendo que eles são iguais. Caso sejam diferentes, informe qual número é o maior, e uma mensagem que são diferentes.*