

# Algoritmos e Programação de Computadores

---

## Algoritmos – Estrutura Sequencial – Parte 1

Professor: Victor Hugo L. Lopes

# *Agenda*

---

- Etapas de ação do computador;
- TDP – Tipos de Dados Primitivos;
- Variáveis;
- Constantes;
- Expressões Aritméticas;
- Instruções Básicas;

# ***Etapas de ação do computador***

---

- O computador independentemente de seu porte e tipo, executa três ações:
  - Entrada de dados;
  - Processamento de dados;
  - Saída de dados;



# *Etapas de ação do computador*

---

- Entrada de dados
  - Computador recebe dados do “mundo externo” e os armazena na memória(?) para uso em algum processamento imediato ou uso em momento posterior oportuno.
- Processamento
  - Utilizando algoritmos o computador transforma dados em informação que pode ser emitida para o “mundo externo”.
- Saída de dados
  - O computador envia dados processados (informação) para o “mundo externo”.
- Para um computador eletrônico funcionar, deve cumprir estas 3 etapas!

# ***Dados primitivos***

---

- Elementos que são manipulados dentro do computador;
- Numéricos → Inteiros e reais;
- Literais → Caracteres ou cadeias;
- Lógicos → Verdadeiros ou falsos;

# ***Dados primitivos***

---

- Numéricos (utilizados em operações matemáticas)
  - Inteiros → Números positivos e negativos sem qualquer valor fracionário (50, 22, 0, -44, -45);
  - Reais → Números positivos e negativos com valores fracionários (-90.554, -32, 0, 77, 8.332);

# ***Dados primitivos***

---

- Literais
  - Representados pelas letras do alfabeto (A – Z e a - z), números (0 – 9) e símbolos (@, #, \$, %, &, \*, (, ), +, ...);
  - Caractere → Delimitado por ‘ ’, utilizado quando se tem apenas um símbolo. Ex.: ‘a’, ‘3’, ‘X’, ‘\$’;
  - Cadeia ou String → Delimitado por “ ”, utilizado quando se tem uma associação de mais de um símbolo. Ex.: “Rua 64 esq. c/ Rua 11”, “Formosa”, “C”, “8”, “-9.99”.

# *Dados primitivos*

---

- Lógicos
  - Valores binários, dicotômicos\*;
  - Sim e Não;
  - Verdadeiro e Falso;
  - True e False;
  - 1 e 0;
  - Booleano → Matemático George Boole (Inglaterra 1815):  
lógica booleana, álgebra booleana



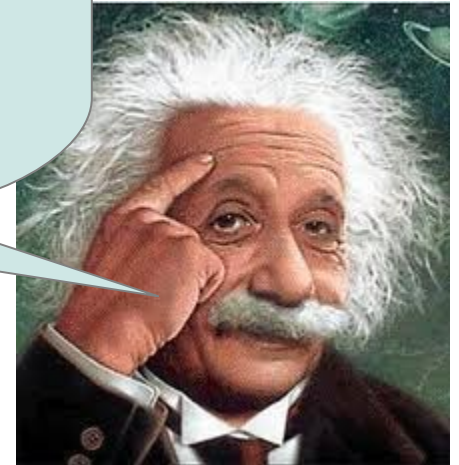
\*bifurcação, bipartição, separação. Distintos.



# ***Dados primitivos***

---

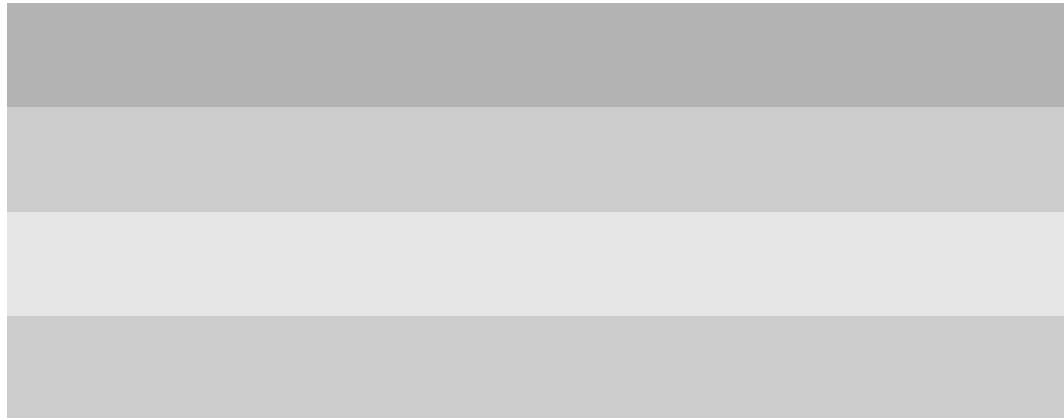
Onde ficam os dados que entraram  
no computador  
Para serem manipulados pelo algoritmo?



# *Variáveis*

---

A memória ram



# *Variáveis*

---

- Tudo que está sujeito a variações, incerto, inconstante, de valor arbitrário;
- Representação de uma região de memória(?) utilizada para armazenar um determinado valor (numérico, literal ou lógico);
- Todo dado a ser armazenado na memória do computador deve ter um tipo!

# *Variáveis*

---

- Analogia com armário com várias gavetas etiquetadas...



# Variáveis

---

- Simplificando: as variáveis na memória do computador devem ter nome e tipo!
- Regras para nomes de variáveis(identificador)
  - Pode ter um ou mais caracteres;
  - O primeiro caractere deve ser alfabético (não são permitidos números, e caracteres especiais);
  - Para nomes compostos, não deve-se utilizar espaço. Deve-se utilizar “\_” (underline);
  - Não pode-se utilizar palavras reservadas da linguagem;
  - Não pode-se utilizar um nome já utilizado;
  - Boa ideia se seu nome for representativo!!!.
- Exemplos de nomes de variáveis?

# Constantes

---

- Assim como variáveis, pode-se definir constantes em um programa.
- Exemplos:
  - $A = B + 1.23$ ;
  - $S = 98 + PI$  (desde que  $PI$  tenha sido definido previamente);
  - Vamos ver melhor na prática!

# Operadores Aritméticos

- Principais operadores aritméticos

Operador	Operação	Descrição
/ (l)	$x / n$	Divisão de “x” por “n”
*	$x * n$	Multiplicação de “x” por “n”
+	$x + n$	Adição de “x” e “n”
-	$x - n$	Subtração de “x” e “n”
←	$x \leftarrow n$	Atribuição do valor “n” a “x”
↑	$x \uparrow n$	Exponenciação de $x^n$
sqrt	sqrt(n)	Raiz Quadrada de “n”
%	$x \% n$	Resto da divisão

# *Expressões Aritméticas*

---

- Processamentos matemáticos;
- Relacionamento entre variáveis e constantes utilizando operadores aritméticos;
- Necessidade de converter expressões matemáticas conhecidas em expressões aritméticas aceitas pelo computador!
- A expressão:

$$x = \{43 \times [55 \div (30 + 2)]\}$$

Deve ser reescrita na forma:

$$x \leftarrow (43 * (55 / (30 + 2)))$$

Abolindo-se assim colchetes e chaves.

Respeito à precedência.



# *Expressões Aritméticas*

---

- A expressão para o cálculo da área de uma circunferência:

$$AREA = \pi \times RAO^2$$

Deve ser reescrita na forma:

$$AREA \leftarrow 3.14159 * RAO \uparrow 2 \text{ ou}$$

$$AREA \leftarrow 3.14159 * RAO * RAO$$

# *Expressões Aritméticas*

---

- A expressão para o cálculo da área de um triângulo:

$$A = \frac{B \times H}{2}$$

Deve ser reescrita na forma:

$$A \leftarrow (B * H) / 2$$

# Expressões Aritméticas

---

- Fórmula de Bhaskara (ou Báskara):

$$x = \frac{-b \pm \sqrt{\Delta}}{2a}$$

$$\Delta = b^2 - 4ac$$

Deve ser adaptada para:

$$\text{Delta} \leftarrow B^2 - 4 * A * C$$

$$X1 \leftarrow (-B + \text{sqrt}(\text{Delta})) / (2 * A)$$

$$X2 \leftarrow (-B - \text{sqrt}(\text{Delta})) / (2 * A)$$

Obs.: X1 e X2 representam x' e x''.

# *Instruções Básicas(comandos)*

---

- Instruções → Representações do conjunto de “palavras chave” de uma linguagem de programação, utilizadas para dar comandos, ordens ao computador;
- Diferentes, porém parecidas nas diversas linguagens (C, C++, Pascal, Java, Basic);
- Instruções em pseudocódigo (português estruturado):
  - início, fim, var, programa, algoritmo, enquanto, se, então, senão, para, escreva, leia, faça, repita, ate\_que...
  - Repare que são ordens no imperativo!

# ***Instruções Básicas(comandos)***

---

- Instruções em LPP para saída de informações do algoritmo:
  - Escreva() ou escreva
- Instruções em LPP para entrada de informações no algoritmo:
  - Leia() ou leia

# *Instruções Básicas(comandos)*

---

- Algumas regras de instruções:
  - Definir os dialetos e palavras reservadas;
  - Codificação manual: letra de forma, cortando o zero para diferenciação do “O”;
  - Declaração de variáveis e constantes no maiúsculo;
  - Todo valor atribuído diretamente no código do algoritmo a uma variável será realizado por meio do símbolo de atribuição (<-);
  - Constantes definidas em db devem receber atribuição de valores com a igualdade (=)
  - Não utilizar símbolos, somente caracteres;
  - Toda referência à variável ou constante serão sempre indicados e utilizados dentro dos símbolos dos diagramas de blocos;
  - Sinal de multiplicação em portugol é o asterisco (\*), em db deve-se utilizar o (x).

# *Instruções Básicas*

---

- Problema básico: programa para efetuar a leitura de dois valores numéricos inteiros, realizar sua soma e apresentar em tela o resultado.
- Qual a sequência de passos para fazermos isso?
- Primeiramente, utilizando o DIA, podemos montar o diagrama de bloco...
- Feito isso, podemos partir para o nosso primeiro pseudocódigo!

# Instruções Básicas

---

algoritmo SOMA\_NUMEROS → Nome do programa

var

X : inteiro

A : inteiro

B : inteiro

Declaração / alocação de variáveis

inicio

leia(A)

leia(B)

Entrada de dados

$X \leftarrow A + B$  → Processamento de dados

escreva(X) → Saída de dados

fimalgoritmo

- Obs. 1: Os nomes de variáveis obedecem às regras descritas;
- Obs 2: As Instruções Básicas (ou palavras reservadas) estão sublinhadas. \_



# *Instruções Básicas*

---

## EXERCÍCIOS

- 1) Desenvolver um algoritmo para receber as horas, os minutos e segundos, calcular e apresentar seu valor em segundos
- 2) Desenvolver um algoritmo para calcular o salário líquido de um professor, que é baseado em salário bruto calculado pelas horas trabalhadas no mês, descontando-se o INSS.
- 3) Desenvolver um algoritmo para ler o dia, mês e ano do nascimento de uma determinada pessoa, e ler o dia, mês e ano atual, calcular e apresentar a idade da pessoa.

# *Instruções Básicas*

---

## EXERCÍCIOS

- 1) Desenvolver um algoritmo para receber as horas, os minutos e segundos, calcular e apresentar seu valor em segundos
- 2) Desenvolver um algoritmo para calcular o salário líquido de um professor, que é baseado em salário bruto calculado pelas horas trabalhadas no mês, descontando-se o INSS.
- 3) Desenvolver um algoritmo para ler o dia, mês e ano do nascimento de uma determinada pessoa, e ler o dia, mês e ano atual, calcular e apresentar a idade da pessoa.