

# Introdução à Computação

Tipos de Dados, Expressões, Operadores e Comandos Básicos

Prof.: Julliano Rosa Nascimento  
julliano.rn@gmail.com



# Roteiro

- Tipos de Dados: Revisão
- Expressões
- Operadores
- Comandos Básicos
- Estrutura de um Programa



# Tipos de Dados

Para que o computador saiba a quantidade de memória a ser reservada para um determinado dado, estes devem ser declarados com seus tipos.

Utilizaremos quatro tipos de dados básicos para representar a informação, na construção de um algoritmo.



# Tipos de Dados

**inteiro**: podem ser positivos, negativos ou nulos.

Exemplo em C:

```
int numero_irmaos;  
numero_irmaos = 15;
```

**real**: podem ser positivos, negativos ou nulos e possuem parte **decimal**.

Exemplo em C:

```
float altura;  
altura = 1.76;
```



# Tipos de Dados

**lógico:** chamados de booleanos e representados por apenas dois valores; verdadeiro (V) e falso (F).

1 - Verdadeiro
0 - Falso

- A porta pode estar ***aberta*** ou ***fechada***.

Exemplo em C:

```
int porta_aberta;  
porta_aberta = 1;
```



# Tipos de Dados

**caractere:** formado por um único caractere ou uma cadeia deles.

- letras maiúsculas e minúsculas;
- números;
- caracteres especiais (&,#,@,?)
- “Universidade Federal de Goiás”
- “João da Silva”
- “nome@gmail.com”

Exemplo em C:

```
char turma = 'B';  
char nome[30] = "Maria";
```



# Constante

- FRASE “Não Fume”
- NOME “Bruna”
- FATOR 1.24
- PI 3.14159

Exemplo em C:

```
#define FATOR 1.24  
#define PI 3.14159
```



# Expressões





# Expressões Aritméticas

São escritas linearmente usando a notação **matemática**, tendo como resposta sempre um valor **numérico**.

Operador	Função	Exemplos
+	adição	$2+3$ , $x+y$
-	subtração	$4-2$ , $n-m$
*	multiplicação	$3*4$ , $a*b$
/	divisão (inteira ou real, conforme operandos)	$10/2$ , $x1/x2$
%	resto da divisão inteira	$16\%3 = 1$



# Funções Matemáticas

Além das operações anteriormente citadas, podemos usar nas expressões aritméticas, **funções** da matemática.



# Funções Matemáticas

Necessário incluir a biblioteca `<math.h>`

Função	Descrição	Tipo do argumento	Tipo do resultado
<code>sin(x)</code>	seno de $x$	real ou inteiro	real
<code>cos(x)</code>	cosseno de $x$	real ou inteiro	real
<code>tan(x)</code>	tangente de $x$	real ou inteiro	real
<code>pow(x,y)</code>	potência $x^y$	real ou inteiro	real
<code>sqrt(x)</code>	raiz quadrada de $x$	real ou inteiro	real
<code>abs(x)</code>	valor absoluto de $x$	real ou inteiro	real ou inteiro
<code>int(x)</code>	converter $x$ em inteiro	real	inteiro



# Funções Matemáticas

O argumento  $x$  pode ser uma constante, variável, expressão aritmética ou outra função matemática.

Exemplos:

$$\text{int}(34.886) = 34$$

$$\text{abs}(-27) = 27$$

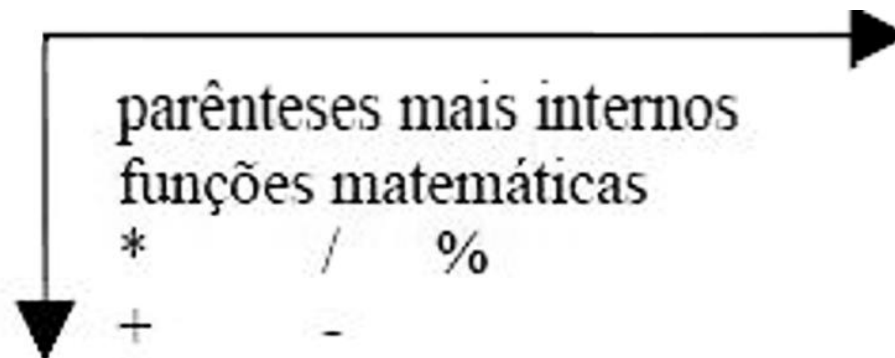
$$\text{pow}(2, 3) = 8$$

$$\text{pow}(\text{sqrt}(9), 3) = 27$$



# Funções Matemáticas

Prioridades: guardam uma **hierarquia** entre si.



# Expressões Lógicas

As expressões lógicas trabalham com operadores **relacionais** ou **lógicos**, tendo como resposta um valor lógico (verdadeiro ou falso).

Exemplos:

15 > 3     **V**

0 <= -1    **F**



# Operadores



# Operadores Relacionais

Operador	Descrição
==	igual a
!=	diferente de
<=	menor ou igual a
>=	maior ou igual a
<	menor que
>	maior que





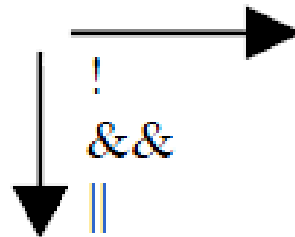
# Operadores Lógicos

Operador	Descrição	Significado
&&	Conjunção	E
	Disjunção	OU
!	Negação	NÃO

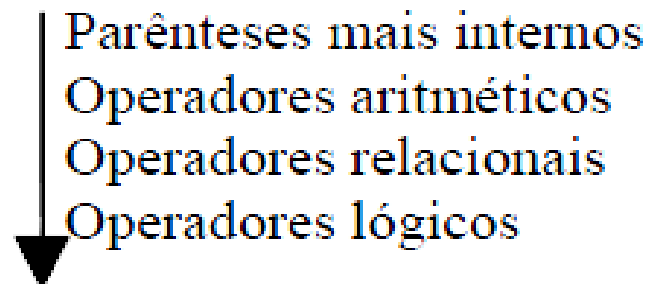


# Prioridades

Entre operadores lógicos:



Entre todos os operadores:



# Comandos Básicos



# Atribuição

Utilizado para **atribuir/designar** valor às variáveis.

O tipo de dado deve ser **compatível** com o tipo da variável.



# Atribuição

Sintaxe:

`<identificador> = <expressão>;`

Exemplos:

```
x = 4;  
turma = 'B';  
cor = "Verde";  
media = (n1+n2)/2;
```

(Lê-se “x recebe 4”)



# Comandos de E/S

Tem como finalidade a interrupção do programa para **entrada** ou **saída** de dados.

Em C, a **entrada** de dados é feita com a função *scanf( )*;

A **saída** de dados é feita com a função *printf( )*;



# LEIA

Não esquecer!

Sintaxe:

```
scanf ("formato", &identificador);
```

Exemplos:

```
scanf ("%d", &inteiro);
```

```
scanf ("%f", &real);
```

```
scanf ("%d/%d/%d", &dia, &mes, &ano);
```



# Especificadores de Formato

<b>Especificador</b>	<b>Tipo de Dado</b>
<code>%c</code>	<code>char</code>
<code>%d</code> ou <code>%i</code>	<code>int</code>
<code>%f</code>	<code>float</code>
<code>%s</code>	uma string (sequência de char)





# IMPRIMA

## Sintaxe:

```
printf ("Mensagem");  
printf ("formato", identificador);
```

## Exemplos:

```
printf ("Ola mundo");  
printf ("x = %d", x);  
printf ("x = %d, y = %d", x, y);
```



# Estrutura Sequencial

Conjunto de **ações** que serão executadas em sequencia linear de forma **horizontal**, na ordem que forem escritas.

As ações serão seguidas por **ponto e vírgula(;)** .

```
float peso;  
printf("Digite o peso:");  
scanf("%f", &peso);
```



# Estrutura de um Programa



# Estrutura de um Programa em C

```
1  #include <stdio.h>
2
3  #define PI 3.14 // constantes
4
5  int main(){
6      float raio, area; // declaracao de variaveis
7
8      printf("Digite o valor do raio: ");
9      scanf("%f", &raio);
10     area = PI*raio*raio;
11     printf("A area do circulo e = %f\n", area);
12
13     return 0;
14 }
```



# Comentários

Um comentário pode ser utilizado em um programa para que as pessoas possam entendê-lo mais facilmente.

Sintaxe:

```
// Comentário ...
```

Exemplo:

```
int num_pecas; // número de peças  
float salario; // salário do funcionário
```



# Exercícios

1) Crie um programa que leia o peso (kg) e a altura (m) de uma pessoa e calcule o seu índice de massa corporal (IMC).

$$\text{IMC} = \frac{\textit{peso}}{\textit{altura}^2}$$

2) Escreva um programa que calcule as raízes de uma equação de segundo grau

$$(ax^2 + bx + c = 0).$$

O programa deverá ler três números ( $a, b, c$ ) e escrever na tela as duas raízes  $x'$  e  $x''$ . Considere somente raízes reais.



# Exercícios

Lista 1 - Comandos Básicos

Exercícios 9, 10 e 11.



# Desafio

Faça um programa para ser utilizado em Caixas Eletrônicos. O cliente deve digitar o valor em reais a ser sacado e o programa deve mostrar quantas cédulas de cada valor (100, 50, 20, 10, 5, 2) e moedas de 1 que serão entregues para o cliente.

Exemplo de Entrada: 288

Exemplo de Saída:

2 notas de 100, 1 nota de 50, 1 nota de 20,  
1 nota de 10, 1 nota de 5, 1 nota de 2, 1 moeda de 1.





# Dúvidas?

