



**Técnicas de Programação I**  
**Prof. Dr. Antonio Marcos SELMINI**  
[profselmini@uol.com.br](mailto:profselmini@uol.com.br)  
[selmini@fiap.com.br](mailto:selmini@fiap.com.br)

**Tipos de dados, Variáveis e Operadores**

## Variáveis

- Quando os dados são digitados para processamento em um programa, onde eles ficam armazenados?
  - Em uma área de armazenamento chamada memória!!
  - A memória é um dispositivo físico onde todos os dados a serem processados ou já processados são armazenados;
  - Basicamente existem dois tipos de memória: primária e secundária;
  - A memória primária é também chamada de RAM (*Random Access Memory*) é uma memória de acesso aleatório e volátil;
  - As memórias secundárias podem ser dispositivos externos tais como: discos magnéticos, CD-ROM, dentre outros;

## Variáveis

- Podemos imaginar a memória principal como uma “caixa”;
- Essa caixa é dividida em caixas menores e dispostas de forma sequencial;
- Essas divisões são chamadas de **posições de memória**;
- Para que o computador faça a manipulação dos dados armazenados é necessário conhecer quais são as posições de memórias que estão preenchidas;
- A cada posição de memória é associado um endereço físico (hexadecimal);

## Variáveis

Endereço Físico	Conteúdo Armazenado
1000:3AC89	“Hoje é feriado”
2000:AAFF3	25

- Conhecer o endereço de memória das variáveis para manipular o seu conteúdo não é nada prático;
- As linguagens de programação permitem que os programadores forneçam nomes às posições de memória;
- Os nomes são chamados de **endereços lógicos** ou **identificadores**;

## Variáveis

- Não podemos esquecer que os nomes das posições de memória são associados aos endereços físicos pelo sistema operacional;

Endereço Lógico	Conteúdo Armazenado
frase	“Hoje é feriado”
idade	25

- As posições de memória que o computador manipula durante a execução de um programa damos o nome de **variáveis**;

## Variáveis

- **Todas as variáveis em um programa deve ter um nome (identificador);**
  - O nome deve ser sugestivo, relacionado com o conteúdo;
  - Nomes de variáveis não podem ser simplesmente números, mas podem conter números;
  - Símbolos especiais não são aceitos (\*, !, ?, etc);
  - Não usar espaços e acentuações;
- **Exemplos:**
  - idade;
  - nome\_do\_funcionario;
  - prova1;
  - 30; **não é um nome válido, pois é um número**
  - prova 1; **não pode haver espaços**

## Tipos de dados primitivos

- **Toda informação manipulada por um programa tem um tipo de dado;**
- **O tipo é importante para que o dado possa ser armazenado corretamente e também indica quais são as operações que podem ser realizadas com a variável;**
- **Os tipos primitivos já vem com a linguagem de programação;**
- **A quantidade de tipos e os nomes podem variar de uma linguagem para outra;**

## Tipos de dados primitivos

- Basicamente os tipos primitivos são;
  - inteiro;
  - real;
  - caractere;
  - cadeia de caracteres ou string;
  - booleano ou lógico;



## Tipos de dados primitivos

- **Inteiro:** valores numéricos sem casa decimal. Exemplo: 12, 100, -12, 0, -56, etc;
  - Operações: adição, subtração, multiplicação, divisão, potenciação e **resto de divisão** (esta operação só é definida para valores inteiros);
- **Real:** valores numéricos com casa decimal. Exemplo: 1.0, 2.456, -5.9090, 0.0, etc ;
  - Operações: adição, subtração, multiplicação, divisão, potenciação;
- **Caractere:** armazenar um único caractere (podendo ser um dígito ou não). Exemplo: '1', 'a', 'A', '\*', '?', '/', ':', etc;

## Tipos de dados primitivos

- **String:** Uma string é formada por dois ou mais caracteres juntos. Exemplo: “aa”, “Maria”, “Pedro”, “112”, etc;
- **Booleano:** toda e qualquer informação que pode assumir apenas duas situações excludentes. São representados pelos valores: **verdadeiro** ou **true** e **falso** ou **false**. Exemplo:
  - A porta pode estar *aberta* ou *fechada*.
  - A lâmpada pode estar *acesa* ou *apagada*.

## Declaração de variáveis

- Toda variável usada em um programa deve ser declarada;
- Declarar uma variável significa fornecer um nome válido e também indicar o tipo de dado que a variável irá armazenar;
- Existem algumas linguagens de programação que não exigem a declaração de variáveis;
- A sintaxe para declarar uma variável é;  
**<tipo\_de\_dado> <nome\_da\_variável>**

## Declaração de variáveis

### ■ Exemplos:

- inteiro idade;
- real nota;
- caractere sexo;
- inteiro x, y, z;
- string nome;
- booleano situacao;

variáveis com o mesmo tipo podem ser declaradas na mesma linha!

evite usar acentuação gráfica

## Constantes

- Um dado é constante quando não sofre nenhuma variação no decorrer do tempo, ou seja, seu valor é constante desde o início até o fim da execução do algoritmo;
- Cada linguagem de programação tem a sua forma de definir constantes. Este assunto será abordado novamente ao estudarmos a linguagem C;
- Exemplo de valores constantes:

$(5 * a + x) / y$

5 é um  
valor  
constante!!

a, x e y são  
variáveis!!

## Operadores

- Mecanismo presente nas linguagens de programação utilizado para manipular o conteúdo das variáveis;
- Basicamente temos quatro grupos de operadores:
  - Atribuição
  - Aritméticos
  - Relacionais
  - Lógicos

## Operador de atribuição

- É responsável por alterar o conteúdo armazenado em uma variável;
- O operador de atribuição varia de uma linguagem de programação para outra;
- O símbolo mais utilizado é o sinal de igual (=), que será o nosso operador de atribuição;
- Vale a pena destacar que em vários livros de algoritmos, o operador de atribuição é representado de forma geral por uma seta ( $\leftarrow$ ). A seta indica a variável que irá receber o conteúdo;

## Operador de atribuição

### ■ Exemplos

- **A = 10** (atribui / armazena o número 10 na variável A);
- **B = 25** (atribui / armazena o número 25 na variável B);
- **C = 'A'** (atribui / armazena o caractere A na variável C);
- **nome = "Maria"** (atribui / armazena a string "Maria" na variável nome);
- **F = A** (atribui / armazena o conteúdo da variável A na variável F);



## Operador de atribuição

- Neste exemplo, as variáveis são declaradas e em seguida inicializadas;

**Início**

**inteiro a, b, c, d**

**a = 2**

**b = 3**

**c = 10**

**d = 5**

**Fim**

## Operador de atribuição

- Uma variável também pode ser inicializada no momento da declaração. Exemplo:

Início

inteiro a = 2, b = 3, c = 10, d = 5

Fim

- É possível também fazer atribuição múltipla, ou seja, um valor pode ser atribuído para várias variáveis. Exemplo:

Início

inteiro a, b, c, d

a = b = 2

c = d = 15

Fim

## Operadores aritméticos

- São responsáveis por realizar as operações aritméticas básicas;

Operação	Símbolo	Exemplo
Adição	+	$A + B$
Subtração	-	$A - B$
Multiplicação	*	$A * B$
Divisão	/	$A / B$

- Estes operadores são binários, ou seja, trabalham com dois operandos;

## Operadores aritméticos

- Além das operações aritméticas básicas, existe também a operação de **resto de divisão**, ou também chamada de **resto da divisão inteira**;
- O operador de resto de divisão varia de uma linguagem de programação para outra. Em C esse operador é representado pelo símbolo de porcentagem (%);
- Essa operação só é definida para variáveis do tipo inteiro. Esse operador é binário. Exemplo:
  - $10 \% 2 = 0$
  - $10 \% 5 = 0$
  - $15 \% 4 = 3$
  - **$7 \% 13 = 7$  não é possível dividir 7 por 13 (divisão inteira)**

## Operadores aritméticos

- Duas outras operações que aparecem com frequência quando trabalhamos com operações aritmética são a potenciação e a radiciação;
- Neste caso não temos operadores e sim funções da linguagem. Neste caso, iremos adotar duas funções da linguagem C: **pow()** (*power*) e **sqrt()** (*square root*);

Operação	Símbolo	Exemplo	Significado
Potenciação	pow()	pow(A, B)	$A^B$
Raiz quadrada	sqrt()	sqrt(A)	$\sqrt{A}$

## Operadores aritméticos

- Os operadores aritméticos têm uma hierarquia que deve ser obedecida:
  - Parênteses;
  - Potenciação ou radiciação;
  - Multiplicação ou divisão ou **resto da divisão**;
  - Adição ou subtração;

## Operadores aritméticos – exercício 1

- Suponha as variáveis  $A = 5$ ,  $B = 1$ ,  $C = 2$ ,  $D = 3$ ,  $E = 0$  e  $F = 1$ . Qual o valor de cada expressão abaixo?
  - a)  $A + B * C$
  - b)  $(A + B) * C$
  - c)  $A - B * C + D * E$
  - d)  $A - B * C + D * E / F$

## Operadores aritméticos – exercício 2

■ Resolva as seguintes expressões:

a)  $5 + 9 + 7 + 8 / 4$

b)  $1 - 4 * 3 / 6 - 3^2$

c)  $5^2 - 4 / 2 + 5 \% 2$

d)  $9 \% 7 \% 3 \% 2 / 5$



## Operadores aritméticos – exercício 3

- Supondo que A, B e C são variáveis do tipo inteiro, com valores iguais a 5, 10 e -8, respectivamente, e uma variável real D, com valor 5.0. Qual o resultado de cada expressão?

a)  $2 * A \% 3 - C$

e)  $-C^2 + (D * 10) / A$

b)  $-2 * C / 4$

f)  $\text{sqrt}(\text{pow}(A, B/A)) + C * D$

c)  $20 / 4 / 4 + 8^2 / 2$

d)  $30 \% 4 * 33 - 1$

## Operadores aritméticos – exercício 4

- Escreva cada uma das expressões abaixo de forma que possam ser processadas por um computador :

$$\text{a) } x = \frac{-b - b^3 - 4ac + 2a^2}{\frac{2a}{(b+1)^2}}$$

$$\text{b) } x = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

$$\text{c) } y = \sqrt[3]{x} + \sqrt[5]{(x+2)^3}$$

$$\text{d) } y = \frac{\sqrt[5]{x^4}}{x-3} + \frac{\frac{2}{3}x^5}{\sqrt{x+y}}$$

## Operadores aritméticos – exercício 5

- Para cada uma das instruções abaixo, escreva a sua expressão aritmética equivalente :

a)  $z = (x + y / 2) / 4 + a + b$

b)  $z = (x + y / 2) / (4 + a + b)$

c)  $z = (x + y / 2) / 4 + a / b$

d)  $z = \text{sqrt}(sp * (sp - a) * (sp - b) * (sp - c))$

## Operadores aritméticos – exercício 6

- Determine quais as codificações corretas para a expressão:

$$x = \frac{-a + b}{2c}$$

- a)  $x = (-a + b) / (2 * c)$
- b)  $x = (b - a) / (2 * c)$
- c)  $x = ((-1) * a + b) / (2 * c)$
- d)  $x = -a + b / 2 * c$
- e)  $x = -a + b / (2 * c)$
- f)  $x = (-a + b) / 2 * c$

## Operadores relacionais

- Os operadores aritméticos são utilizados para realizar cálculos matemáticos com o conteúdo das variáveis;
- Os operadores relacionais como o próprio nome indica são utilizados para relacionar o conteúdo das variáveis;
  - O conteúdo de uma variável é igual ou diferente do conteúdo de outra variável?
  - O conteúdo de uma variável é maior ou menor do que o conteúdo de outra variável?
  - Os operadores relacionais são: >, <, >=, <=, == e !=;
- O resultado de uma expressão relacional é um valor lógico: **VERDADEIRO** ou **FALSO**;

## Operadores relacionais

Operação	Símbolo	Exemplo	Significado
Igualdade	==	A == B	O valor de A é igual ao valor de B?
Maior	>	A > B	O valor de A é maior do que o valor de B?
Maior ou igual	>=	A >= B	O valor de A é maior ou igual ao valor de B?
Menor	<	A < B	O valor de A é menor do que o valor de B?
Menor ou igual	<=	A <= B	O valor de A é menor ou igual ao valor de B?
Diferente	!=	A != B	O valor de A é diferente do valor de B?

- Quando em uma expressão temos os operadores aritméticos junto com os operadores relacionais, a prioridade é dos operadores aritméticos;

## Operadores relacionais – exercício

- Para resolver esse exercício considere a existência de três variáveis do tipo inteiro com seus respectivos conteúdos:  $A = 6$ ,  $B = 3$  e  $C = 2$ .
  - a)  $(A + B / C) < (C / A + 4)$
  - b)  $((B \% C) + 3) >= (B \% A) + (C \% B)$
  - c)  $((B \% C) + 3) <= (B \% A) + (C \% B)$
  - d)  $A * B == A + B - 3 + C / 4$
  - e)  $A / B \% B \% A \% C != B \% A \% B \% C \% A$

## Operadores lógicos

- Os operadores lógicos são extremamente importantes no desenvolvimento de programas (criação de condições lógicas);
- Normalmente temos várias expressões aritméticas e relacionais e devemos combinar os seus resultados com o objetivo de obter apenas um;
- Os operadores lógicos são utilizados para esse propósito;
- O resultado de uma expressão que utiliza operadores lógicos é booleano (VERDADEIRO ou FALSO);



## Operadores lógicos

- Vamos utilizar três operadores lógicos: **E**, **OU** e **NÃO**;
- O operador **E** e **OU** são binários, enquanto que, o operador **NÃO** é unário;

<b>p</b>	<b>q</b>	<b>p E q</b>	<b>NÃO(p E q)</b>	<b>p OU q</b>	<b>NÃO(p OU q)</b>
Verdadeiro	Verdadeiro	Verdadeiro	Falso	Verdadeiro	Falso
Verdadeiro	Falso	Falso	Verdadeiro	Verdadeiro	Falso
Falso	Verdadeiro	Falso	Verdadeiro	Verdadeiro	Falso
Falso	Falso	Falso	Verdadeiro	Falso	Verdadeiro

## Operadores lógicos – exercício

- Determine o resultado de cada uma das expressões abaixo. Considere que as variáveis a e b são do tipo inteiro e armazenam os valores 2 e 7 respectivamente. A variável c é do tipo real e armazena o valor 3,5. A variável d é do tipo lógica e armazena o valor **falso**.
  - a)  $b == a * c \text{ E } (d \text{ ou } b != c)$
  - b)  $b > a \text{ ou } b == \text{pow}(a, a)$
  - c)  $d \text{ E } b / a >= c \text{ OU } \text{NÃO}(a <= c)$
  - d)  $\text{NÃO}(d) \text{ E } \text{sqrt}(a+b) >= c$
  - e)  $b / a == c \text{ OU } b / a != c$
  - f)  $d \text{ OU } \text{pow}(b, a) <= c * 10 + a * b$

## Exercício geral – teste de mesa

- Considere o seguinte trecho de um algoritmo :

inteiro pig, vari, total, a, i

real valor\_a, x

vari = 2

total = 10

valor\_a = 7.0

a = 4

i = 80

x = 4.0

## Exercício geral – teste de mesa

- Informe os valores armazenados em cada uma das variáveis após a execução de cada uma das operações abaixo:

**x = total / vari**

**x = x + 1**

**a = a + i**

**pig = 10**

**a = i / pig**

**a = a + i % 6**

**valor\_a = pig \* valor\_a + x**

## Bibliografia

- DEITEL, Paul; DEITEL, Harvey. C – Como Programar. 6ª edição. São Paulo: Pearson Prentice Hall, 2011.
- FORBELLONE, André Luiz Villar; EBERSPÄCHER, Henri Frederico. Lógica de Programação. 3ª edição. São Paulo: Pearson Prentice Hall, 2005.