

## 2 TIPOS DE DADOS, VARIÁVEIS E EXPRESSÕES

Todo o trabalho realizado por um computador é baseado na manipulação das informações contidas em sua memória. Estas informações podem ser classificadas em dois tipos:

- As **instruções**, que comandam o funcionamento da máquina e determinam a maneira como devem ser tratados os dados.
- Os **dados** propriamente ditos, que correspondem à porção das informações a serem processadas pelo computador.

### 2.1 TIPOS DE DADOS

A classificação apresentada a seguir não se aplica a nenhuma linguagem de programação específica; pelo contrário, ela sintetiza os padrões utilizados na maioria das linguagens.

#### 2.1.1 Inteiros

São caracterizados como tipos inteiros, os dados numéricos positivos ou negativos. Excluindo-se destes qualquer número fracionário. Como exemplo deste tipo de dado, tem-se os valores: 35, 0, -56, 1024 entre outros.

#### 2.1.2 Reais (Ponto flutuante)

São caracterizados como tipos reais, os dados numéricos positivos e negativos e números fracionários. Como exemplo deste tipo de dado, tem-se os valores: 35, 0, -56, 1.2, 1.0, -45.987, 12.67 entre outros.

#### 2.1.3 Caracteres

São caracterizados como tipos caracteres, as sequências contendo letras, números e símbolos especiais. Uma sequência de caracteres deve ser indicada entre aspas (") ou apóstrofes ('), dependendo da linguagem de programação. Este tipo de dado também é conhecido como alfanumérico, *string*, literal ou cadeia. Como exemplo deste tipo de dado, tem-se os valores: "Programação", "UESPI", "Rua Alfa, 52 Apto 1", "Fone 574-9988", "04387- 030", " ", "7", "A", "(86)9453-0098" entre outros.

### 2.1.4 Lógicos

São caracterizados como tipos lógicos os dados com valor **verdadeiro** e **falso**, sendo que este tipo de dado poderá representar apenas um dos dois valores. Ele é chamado por alguns de **tipo booleano**, devido à contribuição do filósofo e matemático inglês George Boole na área da lógica matemática.

## 2.2 VARIÁVEIS

Uma variável pode ser vista como uma caixa com um rótulo ou nome colado a ela, que num dado instante guarda um determinado objeto. O conteúdo desta caixa não é algo fixo, permanente. Na verdade, essa caixa pode ter seu conteúdo alterado diversas vezes. Contudo, o conteúdo deve ser sempre do mesmo tipo.

Na figura 2.1, a caixa (variável) rotulada com **FATOR** contém o valor 5. Como seu tipo é numérico, em um determinado instante essa caixa poderá conter qualquer valor numérico (inteiro ou fracionário; positivo, negativo ou zero). Entretanto, em um determinado instante, ela conterá um, e somente um, valor.



Figura 2.1 – Uma caixa (variável) rotulada com **FATOR** que contém o valor 5.

Variáveis são palavras que tem um significado bem específico em um algoritmo. Para que o computador possa executar comandos que envolvem variáveis da maneira correta, ele deve conhecer os detalhes das variáveis que pretendemos usar. Esses detalhes são: o identificador desta variável e o tipo de valores que essa variável irá conter.

Precisamos assim, de uma maneira de especificar esses detalhes e comunicá-los ao computador. Assim, por exemplo, num algoritmo para calcular a área de um triângulo retângulo pelo Teorema de Pitágoras ( $A^2 = B^2 + C^2$ ), os identificadores A, B e C podem representar as posições de memória que armazenam o valor da hipotenusa (A) e dos catetos (B e C).

É importante que nunca usemos uma palavra reservada, isto é, que faça parte da

linguagem algorítmica, como um identificador, pois poderá causar ambiguidade no entendimento do algoritmo.

### 2.2.1 Declaração de Variáveis

Consiste na definição dos nomes e valores das constantes e dos nomes e tipos das variáveis que serão utilizadas pelos algoritmos, previamente à sua utilização, incluindo comentário, quando se fizerem necessários.

Na maioria das linguagens de programação, quando o computador está executando um programa e encontra uma referência a uma variável ou a uma constante qualquer, se esta não tiver sido previamente definida, ele não saberá o que fazer com ela. Da mesma forma, um programador que estiver implementando um algoritmo, em alguma linguagem de programação, terá o seu trabalho simplificado se todas as constantes e variáveis referenciadas no algoritmo tiverem sido previamente declaradas. As constantes são declaradas antes das variáveis. Vejamos os formatos da declaração e alguns exemplos.

O significado da declaração de variáveis corresponde à criação de locais na memória rotulada com o nome da variável (identificador) e marcada com o tipo de valores que ela pode conter. Para que os programas manipulem valores, estes devem ser armazenados em variáveis e para isso, devemos declará-las de acordo com a sintaxe:

**identificador1, identificador2, ... ,identificadorN : tipo;**

Vejamos os exemplos a seguir:

*X1 : inteiro; obs.: X1 é o nome de um local de memória que só pode conter valores inteiros.*

SOMA, MÉDIA : real;

frase, nome : caractere;

PAR, Resto : inteiro;

A, B : real;

TEM, Achou : lógico;

## 2.3 OPERADORES

**Operadores** são elementos fundamentais que atuam sobre **operandos** e produzem um determinado resultado. Por exemplo, a expressão **3 + 2** relaciona dois operandos (os números 3 e 2) por meio do operador (+) que representa a operação de adição.

De acordo com o número de operandos sobre os quais os operadores atuam, os últimos podem ser classificados em:

- **Binários**, quando atuam entre dois operandos. Esta operação é chamada diádica. Ex.: os operadores das operações aritméticas básicas (soma, subtração, multiplicação e divisão).
- **Unários**, quando atuam sobre um único operando. Esta operação é chamada monádica. Ex.: o sinal de (-) na frente de um número, cuja função é inverter seu sinal.

Outra classificação dos operadores é feita considerando-se o tipo de dado de seus operandos e do valor resultante de sua avaliação. Segundo esta classificação, os operandos dividem-se em **aritméticos, lógicos e literais**. Esta divisão está diretamente relacionada com o tipo de expressão onde aparecem os operadores. Um caso especial é o dos operadores **relacionais**, que permitem comparar pares de operandos de tipos de dados iguais, resultando sempre num valor lógico.

### 2.3.1 Operadores de Atribuição

Um operador de atribuição serve para atribuir um valor a uma variável. Em Algoritmo usamos o operador de atribuição: `:=` A sintaxe de um comando de atribuição é:

**Nome\_da\_Variável := Expressão;**

A **expressão** localizada no lado direito do sinal dois pontos e igual (**atribuição**) é avaliada e o valor resultante é armazenado na variável à esquerda. O nome da variável aparece sempre sozinho, no lado esquerdo do sinal de atribuição deste comando.

### 2.3.2 Operadores Aritméticos

Os operadores aritméticos representam as operações mais comuns da matemática, ou seja, os operadores aritméticos se relacionam às operações aritméticas básicas, conforme a tabela a seguir.

A prioridade entre operadores define a ordem em que os mesmos devem ser avaliados dentro de uma mesma expressão.

Operador	Tipo	Operação	Prioridade
+	Binário	Adição	4
-	Binário	Subtração	4
*	Binário	Multiplicação	3
/	Binário	Divisão	3
MOD	Binário	Resto da Divisão	3
DIV	Binário	Divisão Inteira	3
**	Binário	Exponenciação	2
+	Unário	Manutenção do Sinal	1
-	Unário	Inversão do Sinal	1

### 2.3.3 Operadores Relacionais

Os operadores relacionais são operadores binários que devolvem os valores lógicos verdadeiro e falso, após avaliar uma expressão.

Operador	Comparação
>	maior que
<	menor que
>=	maior ou igual
<=	menor ou igual
=	igual
<>	diferente

Estes valores são somente usados quando se deseja efetuar comparações. Comparações só podem ser feitas entre objetos de mesma natureza, isto é, variáveis do mesmo tipo de dado. O resultado de uma comparação é sempre um valor lógico.

### 2.3.4 Operadores Lógicos

Os operadores lógicos ou booleanos são usados para combinar expressões relacionais. Também devolvem como resultado valores lógicos verdadeiro ou falso.

Operador	Tipo	Operação	Prioridade
OU	Binário	Disjunção	3
E	Binário	Conjunção	2
NÃO	Unário	Negação	1

### 2.3.5 Operadores Literais

Os operadores que atuam sobre caracteres variam muito de uma linguagem para outra. O operador mais comum e mais usado é o operador que faz a concatenação de strings: toma-se duas strings e acrescenta-se (concatena-se) a segunda ao final da primeira.

O operador que faz esta operação é: +

Por exemplo, a concatenação das strings “ALGO” e “RITMO” é representada por:

**“ALGO” + ”RITMO”**

e o resultado de sua avaliação é: “ALGORITMO”

## 2.4 EXPRESSÕES

O conceito de expressão em termos computacionais está intimamente ligado ao conceito de expressão ou fórmula matemática, onde um conjunto de variáveis e constantes numéricas relacionam-se por meio de operadores aritméticos compondo uma fórmula que, uma vez avaliada, resulta num valor.

### 2.4.1 Expressões Aritméticas

Expressões aritméticas são aquelas cujo resultado da avaliação é do tipo numérico, seja ele inteiro ou real. Somente o uso de operadores aritméticos, variáveis numéricas e parênteses é permitido em expressões deste tipo.

Exemplos:

Expressão	Resultado
1 + 2	3
5.0 - 1	4.0
2 * 1.5	3.0
5 / 2	2.5
5 DIV 2	2
5 MOD 2	1

### 2.4.2 Expressões Lógicas

Expressões Lógicas são aquelas cujo resultado da avaliação é um valor lógico verdadeiro ou falso.

Nestas expressões são usados os operadores relacionais e os operadores lógicos, podendo ainda serem combinados com expressões aritméticas.

Quando forem combinadas duas ou mais expressões que utilizem operadores relacionais e lógicos, os mesmos devem utilizar os parênteses para indicar a ordem de precedência.

A seguir, exemplos de expressões relacionais:

Expressão	Resultado
$1 = 2$	FALSO
$'A' = 'a'$	FALSO
$5 > 2$	VERDADEIRO
$3 \leq 3$	VERDADEIRO
$\text{VERDADEIRO} < \text{FALSO}$	FALSO
$'JOAO' > 'JOSE'$	FALSO
$2 + 3 \diamond 5$	FALSO
$'comp' \diamond 'COMP'$	VERDADEIRO
$'11' < '4'$	VERDADEIRO

A tabela verdade (abaixo) apresenta o resultado de cada operador lógico, com os valores dados para as expressões lógicas A e B:

A	B	A e B	A ou B	não A	não B
VERDADEIRO	VERDADEIRO	VERDADEIRO	VERDADEIRO	FALSE	FALSE
VERDADEIRO	FALSE	FALSE	VERDADEIRO	FALSE	VERDADEIRO
FALSE	VERDADEIRO	FALSE	VERDADEIRO	VERDADEIRO	FALSE
FALSE	FALSE	FALSE	FALSE	VERDADEIRO	VERDADEIRO

Uma expressão relacional ou lógica retornará falso para o valor lógico falso e verdadeiro para o valor lógico verdade.

Como a ordem de precedência dos operadores lógicos é maior que a dos operadores relacionais, devem sempre ser usados parênteses quando se escrever uma expressão lógica complexa. Por exemplo:

$$(A > B) \text{ OU } (B = C)$$

### 2.4.3 Expressões Literais

Expressões literais são aquelas cujo resultado da avaliação é um valor literal (caractere). Neste tipo de expressões só é usado o operador de literais (+).

### 2.4.4 Avaliação de Expressões

Expressões que apresentam apenas um único operador podem ser avaliadas diretamente. No entanto, à medida que as mesmas vão tornando-se mais complexas com o aparecimento de mais de um operando na mesma expressão, é necessária a avaliação da mesma passo a passo, tomando um operador por vez. A sequência destes passos é definida de acordo com o formato geral da expressão, considerando-se a prioridade (precedência) de avaliação de seus operadores e a existência ou não de parênteses na mesma.

As seguintes regras são essenciais para a correta avaliação de expressões:

1. Deve-se observar a prioridade dos operadores, conforme mostrado nas tabelas de operadores: operadores de maior prioridade devem ser avaliados primeiro. Se houver empate com relação à precedência, então a avaliação se faz da esquerda para a direita.
2. Os parênteses usado em expressões tem o poder de “roubar” a prioridade dos demais operadores, forçando a avaliação da sub-expressão em seu interior.
3. Entre os quatro grupos de operadores existentes, a saber, aritmético, lógico, literal e relacional, há certa prioridade de avaliação: os aritméticos e literais devem ser avaliados primeiro; a seguir, são avaliadas as sub-expressões com operadores relacionais e, por último os operadores lógicos são avaliados.

## 2.5 EXERCÍCIOS PROPOSTOS

1. Qual o resultado das expressões aritméticas abaixo, sabendo-se que os valores de X, Y e Z são, respectivamente, 1, 2 e 5 ?
  - A)  $Z \bmod Y \div Y$
  - B)  $X + Y + Z / 3$
2. Preencha a Tabela Verdade abaixo:

A	B	A and B	A or B	not A	not B
TRUE	TRUE				
TRUE	FALSE				
FALSE	TRUE				
FALSE	FALSE				