

Linguagem e Técnica de Programação 1 – (LTP1)

CAPÍTULO – 2

DADOS

E

INSTRUÇÕES

Prof. Arnaldo

SUMÁRIO

- Dados
 - Tipos de dados
 - Inteiro
 - Real
 - Caracter
 - Lógico
- Variáveis
- Constantes
- Instruções
 - Entrada, Processamento e Saída
 - Representação
 - Fluxograma
 - Portugol

DADOS

- Após a **introdução a lógica de programação**, neste capítulo será apresentado uma aplicação mais prática sobre algoritmos em **portugol** e **fluxogramas**, para que possamos entender que o desenvolvimento de programas para o computador tem como objetivo solucionar um problema que envolva a manipulação de **informações**.
- Essas **informações** podem ser divididas em dois tipos básicos: **dados** e **instruções**.

- **Definição**
 - ❑ Os dados são, na verdade, os valores que serão utilizados para a resolução de um problema.

TIPOS DE DADOS

Dados são **informações** que serão processadas pelo computador.

Essas **informações** podem ser classificadas em 04 tipos de dados primitivos:

- ✓ Inteiro
- ✓ Real
- ✓ Caracter
- ✓ Lógico

Tipo: Inteiro

Os dados classificados como inteiro são aqueles numéricos positivos ou negativos, com exceção dos números fracionários.

Exemplo 1:

| Tipo Inteiro |
|--------------|
| 54 |
| -89 |
| 0 |
| 13 |

Exemplo 2:

- Na sala existem 10 cadeiras.
- Eu comprei 5 calças.

Tipo: Real

O tipo real compreende os números positivos, negativos e fracionários.

Exemplo 1:

| Tipo Real |
|-----------|
| 54 |
| -89 |
| 0 |
| 0,13 |

Exemplo 2:

- O piso salarial do setor de criação é R\$ 703,56.
- A largura da cama é 0,88 cm.

Tipo: Caractere

Os dados classificados como caractere são as sequências contendo letras, números e símbolos especiais (@, !, #, \$, %, & , *). Essa sequência de caracteres deve ser delimitada por aspas (“ ”) ou apóstrofos (‘ ’).

Exemplo 1:

| Tipo Caractere |
|-----------------------|
| “Rua das Flores, 145” |
| “Aviso” |
| “CEP: 18900-00” |
| “10” |

Exemplo 2:

- Entregue na “Rua das Flores, 145”.
- Lê-se naquela placa “Lave os utensílios após o uso”.

Tipo: Lógico

Os dados classificados como lógico são aqueles que podem assumir somente dois valores: **Verdadeiro** ou **Falso**.

Exemplo:


- O zíper pode estar aberto ou fechado.
- O computador pode estar ligado ou desligado.

Variáveis

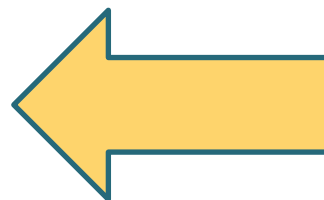
O termo **variável** tem como definição aquilo que é sujeito a mudança, que é incerto, instável ou inconstante. Em programação a quantidade de informações que serão processadas é muito grande e diversificada, com isso, os dados se tornam bastante variáveis.

A **memória** de um computador pode ser **comparada** com um grande **arquivo** com inúmeras **gavetas**, que seriam os **locais físicos** responsáveis por **armazenar dados**. Cada gaveta pode armazenar um único valor de um tipo específico por vez. Considerando que a memória de um computador seria um arquivo muito grande, é necessário **identificar cada gaveta** com uma etiqueta para facilitar a localização da informação.

A **imagem** a seguir representa a
comparação comentada



Variáveis



As variáveis são como as gavetas do arquivo, onde cada gaveta corresponde a uma posição de memória onde será armazenada a informação. Apesar de uma variável poder assumir diferentes valores, ela só pode armazenar um valor de cada vez.

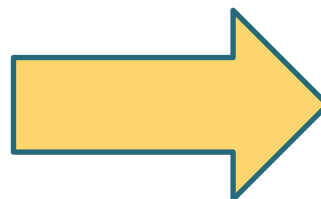
Toda variável é identificada por um nome ou identificador para sua localização e posterior uso dentro de um programa.

Variáveis

As **regras** para nomear uma variável são:

- O nome pode ter um ou mais caracteres.
- O primeiro caractere deve ser sempre uma letra.
- Pode utilizar apenas letras e números.
- Não pode conter caracteres especiais.
- Não pode possuir espaços em branco.
- Não pode ser uma palavra reservada, ou seja, que faça parte da linguagem de programação utilizada.

Veja como ficaria a declaração de variáveis em um algoritmo:



Exemplos

Nomes **válidos**:

- nome, numero1, A, N2.

Nomes **inválidos**:

- nome usuario, 2num, email@, var.

```
algoritmo <Nome do programa>  
var  
    nome: CARACTER  
    preco: REAL  
    numero: INTEIRO  
inicio  
    <Comandos>  
fimalgoritmo
```

Constantes

- O termo significa que algo é inalterável. Em um programa de computador as constantes são endereços de memória que guardam valores fixos, ou seja, não se alteram ao longo da execução de um programa.
- São classificadas de acordo com o tipo de dado que armazena, a saber: numérica, lógica ou literal.

Exemplos:

Fórmula de Baskara:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

2 é uma constante.

Volume do Cilindro:

$$V_{\text{cilindro}} = \pi \cdot r^2 \cdot h$$

Π (pi) é uma constante (3,14).

Instruções

As instruções, **ou comandos**, são representados por um conjunto de palavras correspondentes a linguagem de programação utilizada, para que o computador possa interpretar e executar as **instruções**. Assim como existem várias línguas diferentes para a comunicação entre os seres humanos, há também inúmeras linguagens de programação que o computador consegue interpretar, como: **Pascal, Delphi, C, PHP, ASP, Java, etc.** Com isso, uma mesma instrução pode ser escrita de forma diferente, dependendo da linguagem utilizada para desenvolvê-la.

Entrada, Processamento e Saída

- Para desenvolver um programa de computador é necessário saber que ele passa por 03 (três) fases de trabalho: **entrada** de dados, o **processamento** desses dados e a **saída** dos mesmos.
- A eficiência do resultado no processo de saída de um programa depende da qualidade na forma de coletar os dados.
- Os erros causados pelo computador são provocados por falha humana, pois um computador não pode errar por vontade própria.
- A execução de um programa é feita da seguinte forma: após a entrada dos dados com a instrução **leia** (em **portugol**), os dados são processados através de uma sequência da manipulação das variáveis e por fim, são exibidos os dados processados com a instrução **escreva** (em **portugol**).
- **Entrada** de dados pode ser feita via modem, teclado, discos, leitores óticos, etc.
- **Saída** de dados pode ser feita em vídeo, impressora, disco entre outras.

Representação

- ✓ Há várias formas de se representar o problema para o qual iremos desenvolver um programa de computador. Estas formas de representação tem como papel fundamental auxiliar o programador ou o analista de sistemas, a entender e resolver o problema, para depois buscar a sua solução dentro de um computador.
- ✓ Na verdade, o que vamos fazer é ensinar o computador a resolver o problema por meio de um programa.
- ✓ O segredo de uma boa lógica de programação está no conhecimento aprofundado do problema a ser solucionado.
- ✓ As 02 (duas) formas de representar as instruções são:
 - **fluxograma.**
 - **portugol.**

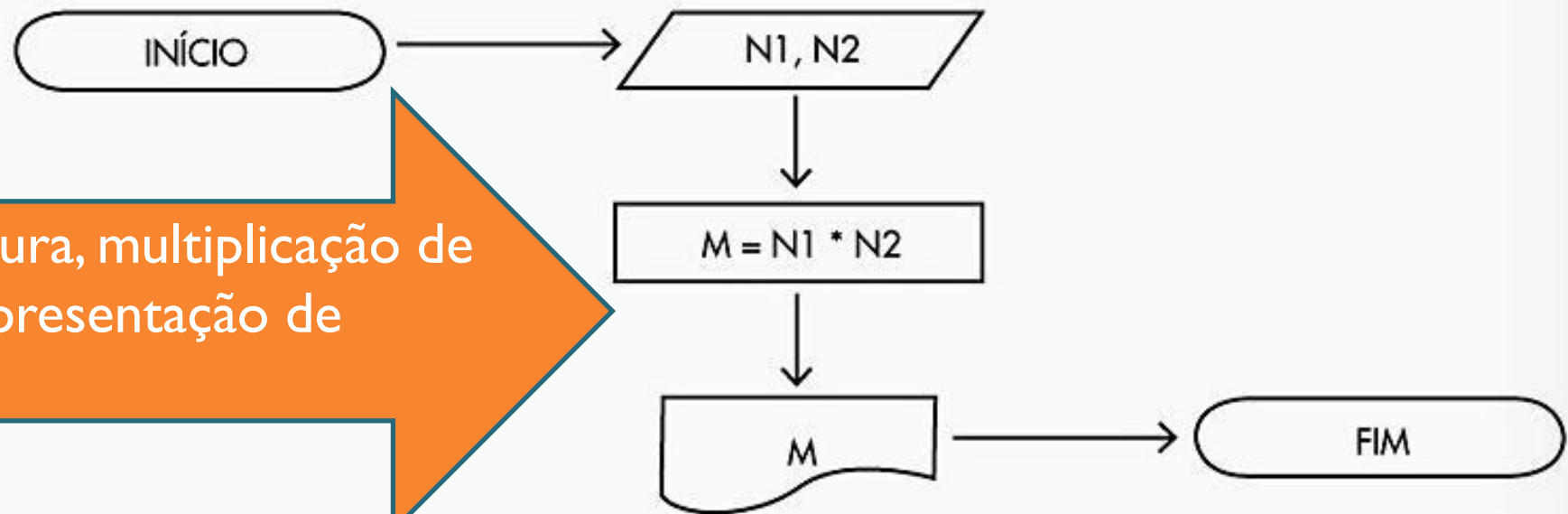
Fluxograma

Tem como objetivo buscar a solução de um problema separando-o em pequenas partes para facilitar a compreensão.

Após interpretar o problema e definir as variáveis, a serem utilizadas, pode-se diagramar o algoritmo com o uso de fluxogramas, que é a forma mais utilizada.

Exemplo:

Fluxograma da leitura, multiplicação de dois números e apresentação de resultados



Portugol

- É uma pseudolinguagem que utilizaremos para desenvolver a lógica de programação simulando uma linguagem de computador.
- A pseudolinguagem “multiplicação” usa palavras comuns ao nosso vocabulário como: **leia**, **escreva**, **inicio** e **fim**.

Exemplo:

```
algoritmo "multiplicacao"  
var  
    m: real  
    n: real  
    x: real  
inicio  
    leia (m)  
    leia (n)  
    x <- m*n  
    escreva (x)  
fimalgoritmo
```

A declaração de variáveis deve ser feita antes da execução das instruções do programa (**início**) do programa. Após ter definido as variáveis, começa a execução das instruções, neste caso, conceder valor as variáveis **M** e **N** através do comando **leia**; **X** recebe o valor da multiplicação de **M** e **N**. Por fim, o resultado da operação armazenado em **X** será exibido na tela do computador através do comando **escreva**.

Referência Bibliográfica

- ❖ PUGA, Sandra e RISSETTI, Gerson. Lógica de Programação Estruturas de Dados, com aplicações em Java. 2.ed. São Paulo: Pearson Prentice Hall, 2009.
- ❖ SILVA, Camila Ceccatto da e PAULA, Everaldo Antônio de. Lógica de Programação: aprendendo a programar. Santa Cruz do Rio Pardo, SP:Viena, 2007.