



# Algoritmos e Estruturas de Dados I

## AED I

Análise e Desenvolvimento de Sistemas

# Análise e Desenvolvimento de Sistemas



# Ementa da disciplina

- Algoritmos e técnicas de programação estruturada:
  - programação sequencial
  - estrutura condicional
  - estrutura de repetição
- Tipos de dados básicos e estruturados
- Listas
- Filas
- Pilhas
- Ponteiros

# Atividades da disciplina

- Exercícios:
  - Muitas listas de exercícios
- Aulas práticas
  - C++

# Atividades da disciplina

## **1º Semestre**

$$\text{Média1} = (\text{Av1} + \text{Av2} + \text{Av3} + \text{Exercícios}) / 3$$

## **2º Semestre**

$$\text{Média2} = (\text{Av4} + \text{Av5} + \text{Av6} + \text{Exercícios}) / 3$$

$$\text{Média\_Final} = (\text{Média1} + \text{Média2}) / 2$$

Se ( $\text{Média\_Final} \geq 7,0$ )

então “Aprovado”

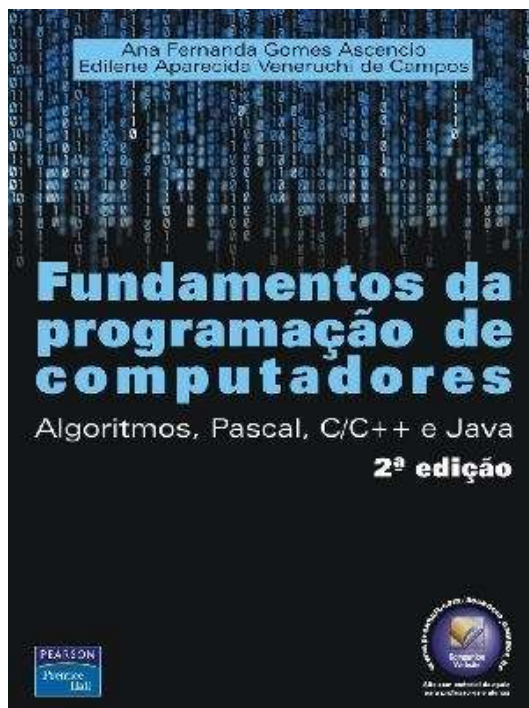
Se ( $\text{Média\_Final} \geq 4,0$ ) E ( $\text{Média\_Final} < 7,0$ )

então “Exame”

# Referências bibliográficas



Ascencio, A.F.G., Campos, E.A.V.  
*Fundamentos da programação de computadores*. São Paulo: Prentice Hall, 2002.



Ascencio, A.F.G., Campos, E.A.V.  
*Fundamentos da programação de computadores*. 2ª Edição. São Paulo: Prentice Hall, 2007.

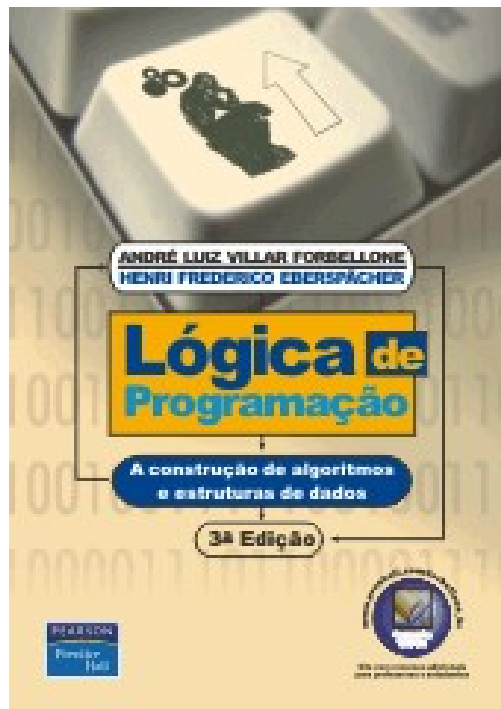




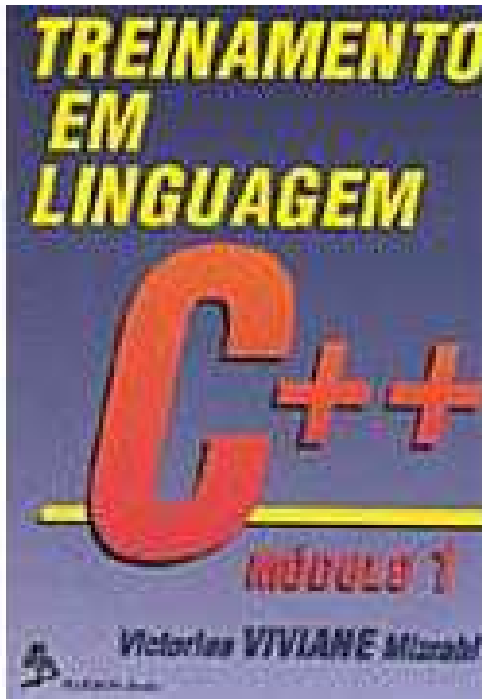
FARRER, H, et al. *Algoritmos estruturados*. Rio de Janeiro: Guanabara, 1989.



Forbellone, A.L.V. *Lógica de Programação*: a construção de algoritmos e estrutura de dados. São Paulo: Makron, 1993.



Forbellone, A.L.V. *Lógica de Programação*: a construção de algoritmos e estrutura de dados. 3ª Edição. São Paulo: Makron, 2005.



MIZRAHI, Victorine V. Treinamento em Linguagem C++: Módulo 1. São Paulo: Makron Books, 1990.



MIZRAHI, Victorine V. Treinamento em Linguagem C++: Módulo 2. São Paulo: Makron Books, 1990.

Não seguro | fema.edu.br

Fundação Lemann | Salman Khan - Kha... | Antonio Artigo copi... | Uma Introdução às... | Introdução às Rede... | Redes Neurais Artifi...

FALE CONOSCO · PORTAL ACADÊMICO · MOODLE

**fema**  
Fundação Educacional do Município de Assis

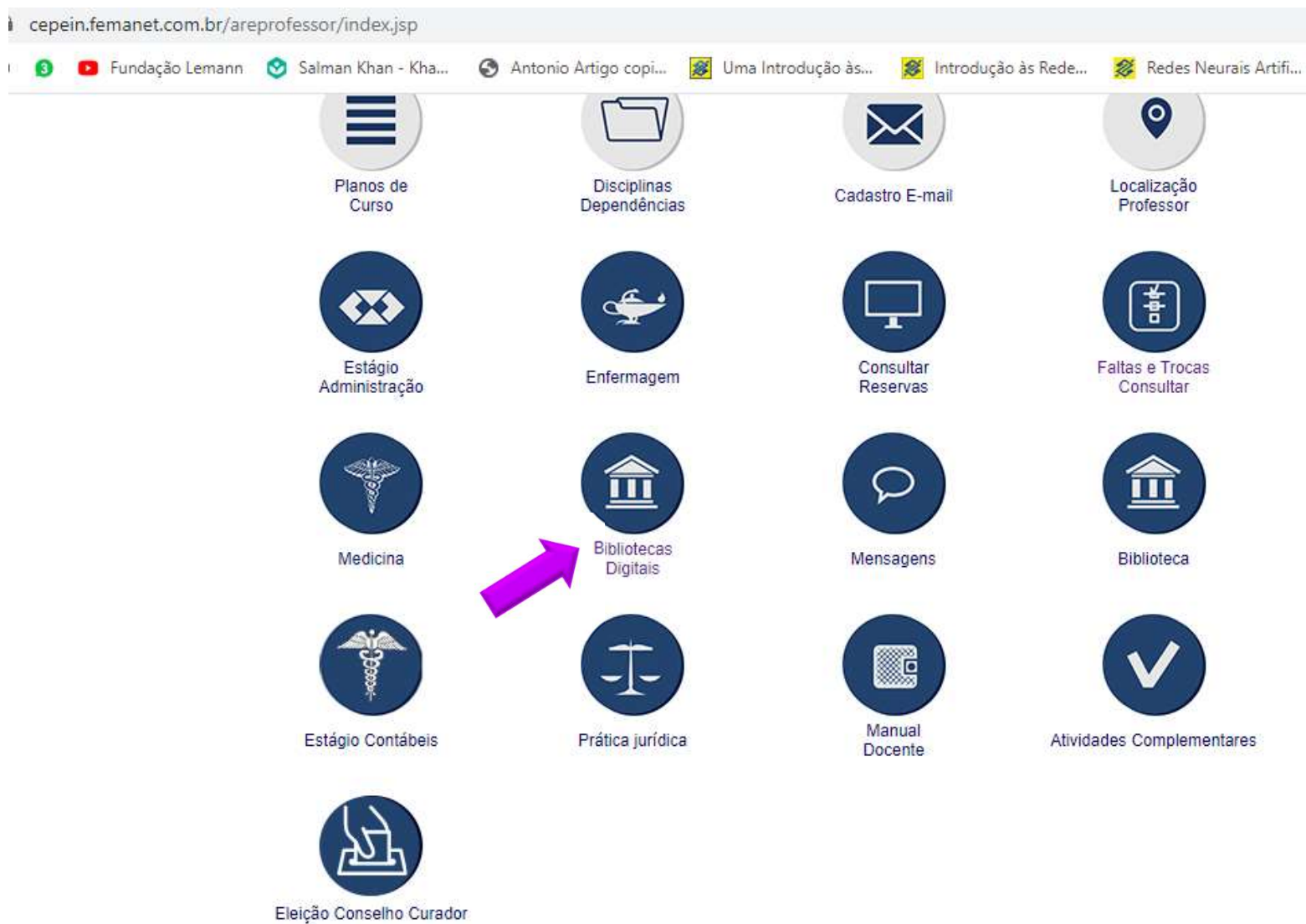
FEMA

GRADUAÇÃO | PÓS-GRADUAÇÃO | PESQUISA | EXTENSÃO | VESTIBULAR | BLOG

**VOLTA ÀS AULAS!**  
CALOUROS 2021

RECADO ESPECIAL AOS CALOUROS  
HOJE 22/02/2021  
AS AULAS SE INICIAM REMOTAMENTE

**fema**  
Fundação Educacional do Município de Assis





[Instru  es de acesso Proview RT](#)



## Correspondências de livro



Algoritmo e programação -  
Engenharia



Complexidade de Algoritmos -  
V13 - UFRGS



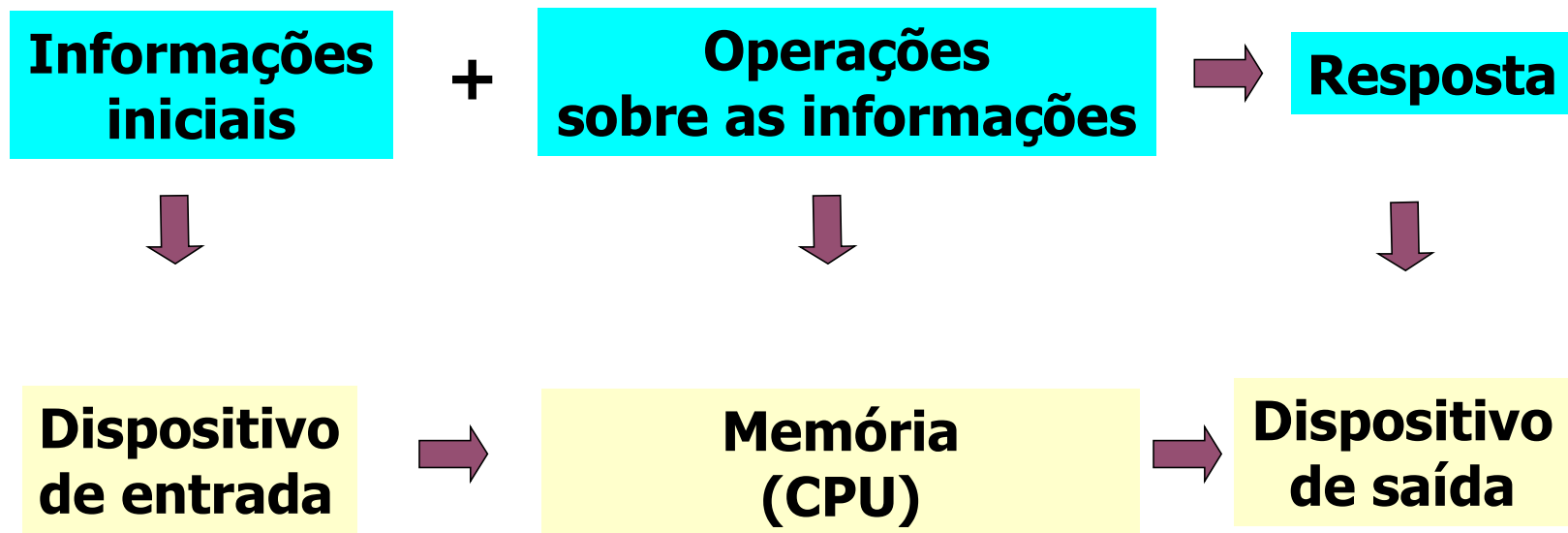
Desmistificando Algoritmos



Algoritmos



# Processamento de Dados



# Portanto...

- Um computador tem duas partes diferentes que trabalham juntas:
  - Hardware: composto pelas partes físicas
  - Software: composto pelos programas

# No entanto...

- Para que o computador compreenda e execute programas, devemos escrevê-los utilizando uma linguagem que tanto o computador quanto o *desenvolvedor* de *software* entendam.
- Essa linguagem é chamada de *linguagem de programação*.

## Etapas p/ o desenvolvimento de um programa...

- **Análise**: estudar o enunciado do problema para definir os dados de entrada, o processamento e os dados de saída;
- **Algoritmo**: ferramenta narrativa utilizada para descrever o problema com suas soluções;
- **Codificação**: o algoritmo é transformado em códigos da linguagem de programação escolhida.

# Conceito de algoritmo...

- “É uma sequência de passos que visa atingir um objetivo bem definido” (Forbellone, 1999)
- “É a descrição de uma sequência de passos que deve ser seguida para a realização de uma tarefa” (Ascencio, 1999)

# Conceito de algoritmo...

- “É uma sequência finita de instruções ou operações cuja execução, em tempo finito, resolve um problema computacional, qualquer que seja sua instância” (Salvetti, 1999)
- “São regras formais para a obtenção de um resultado ou da solução de um problema, englobando fórmulas de expressões aritméticas” (Manzano, 1997)

# Algoritmo 1 – Somar 3 números

- Passo 1 – Receber os três números
- Passo 2 – Somar os três números
- Passo 3 – Mostrar o resultado obtido



## Algoritmo 2 – Fazer um sanduíche

- Passo 1 – **Pegar** o pão.
- Passo 2 – **Cortar** o pão ao meio.
- Passo 3 – **Pegar** a maionese.
- Passo 4 – **Passar** a maionese no pão.
- Passo 5 – **Pegar** e **cortar** alface e tomate.
- Passo 6 – **Colocar** alface e tomate no pão.
- Passo 7 – **Pegar** o hamburguer.
- Passo 8 – **Fritar** o hamburguer.
- Passo 9 – **Colocar** o hamburguer no pão.

# Exercícios

- Algoritmo 3 – Trocar uma lâmpada
- Algoritmo 4 – Fazer um suco de laranja

# Método p/ construção de algoritmos

- Ler atentamente o enunciado;
- Definir os dados de entrada, ou seja, quais dados serão fornecidos;
- Definir o processamento, ou seja, quais cálculos serão efetuados e quais as restrições para esses cálculos. O processamento é responsável pela transformação dos dados de entrada em dados de saída.

## Método p/ construção de algoritmos (cont.)

- Definir os dados de saída, ou seja, quais dados serão gerados depois do processamento;
- Construir o algoritmo;
- Testar o algoritmo realizando simulações (Teste de Mesa).

# Exercícios



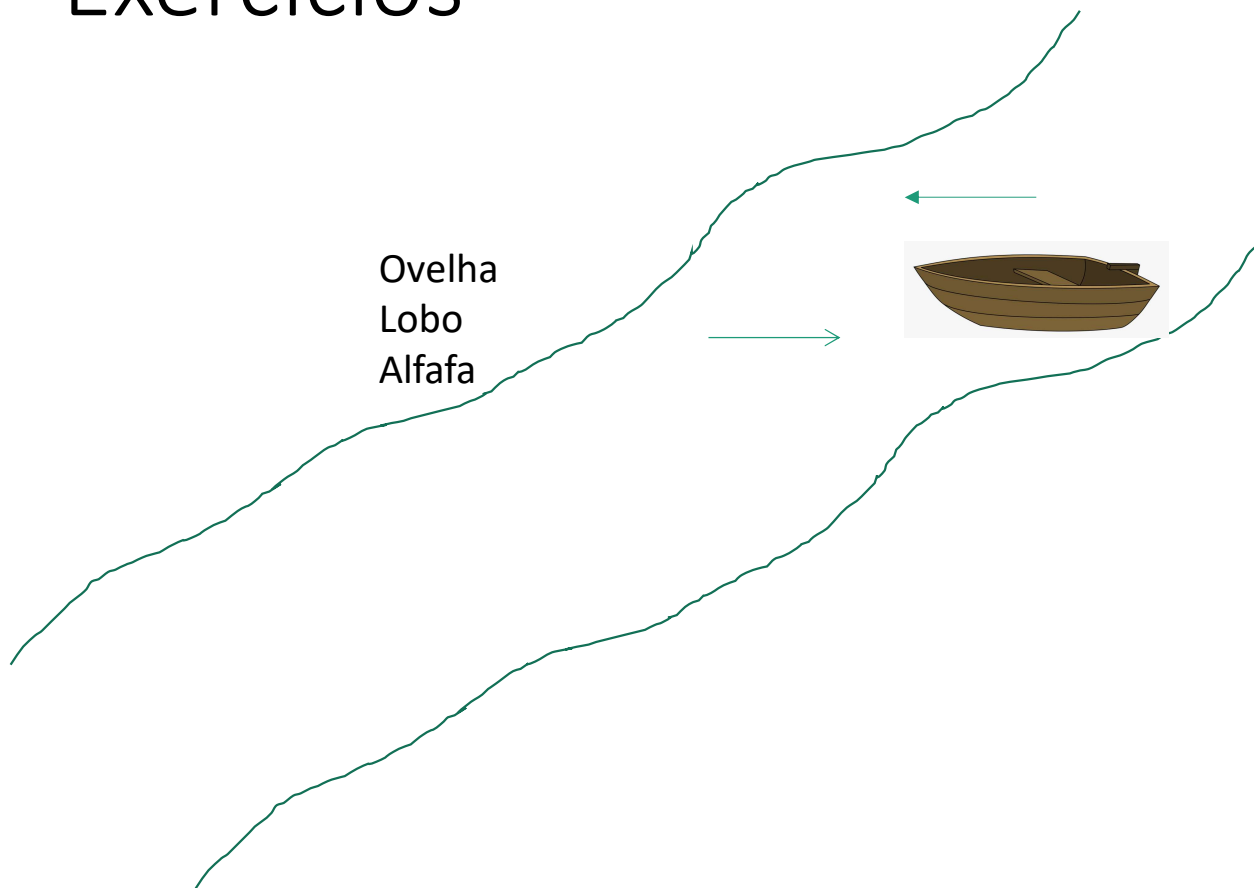
- Um homem precisa atravessar um rio com um barco que possui capacidade de carregar apenas ele mesmo e mais uma de suas três cargas: um lobo, uma cabra e um maço de alfafa.
- O que o homem deve fazer para conseguir atravessar o rio sem perder suas cargas?

# Exercícios

- Pontos a considerar:
  - Lobo come a ovelha
  - Ovelha come a alfafa



# Exercícios



Ovelha  
Lobo  
Alfafa

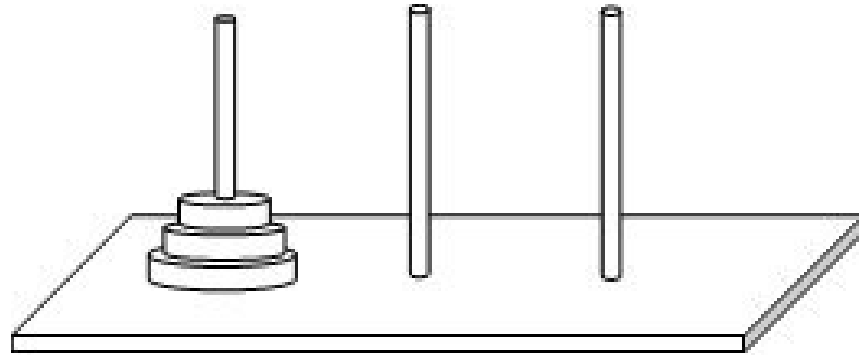


# Exercícios - Solução

1. Homem atravessa com a Ovelha
2. Homem retorna sozinho
3. Homem atravessa com a Alfafa
4. Homem retorna com a Ovelha
5. Homem atravessa com o Lobo
6. Homem retorna sozinho
7. Homem atravessa com a Ovelha
8. Fim.

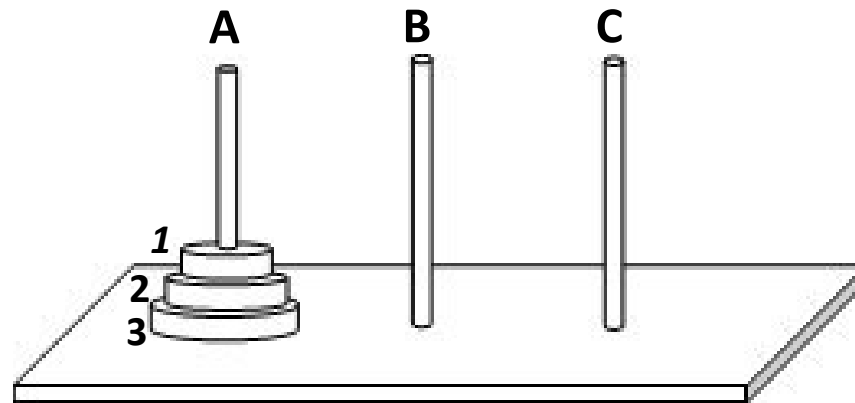


## Exercícios (cont)



- A Torre de Hanoi consiste em três hastes (**a**, **b** e **c**), uma das quais serve de suporte para três discos de tamanhos diferentes (1 – 2 – 3), os menores sobre os maiores. Pode-se mover um disco de cada vez para qualquer haste, contanto que nunca seja colocado um disco maior sobre um disco menor. O objetivo é transferir os três discos para outra haste.

# Exercícios (cont)



# Tipos de algoritmos

- Descrição Narrativa
- Fluxograma
- Pseudocódigo ou Portugol

# Descrição Narrativa

- Consiste em analisar o enunciado do problema e escrever, **utilizando uma linguagem natural**, os passos a serem seguidos p/ a resolução do problema.
- **Desvantagem:** a linguagem natural abre espaço p/ várias interpretações, o que posteriormente dificultará a transcrição desse algoritmo para programa.

# Fluxograma

- Consiste em analisar o enunciado do problema e escrever, **utilizando símbolos gráficos predefinidos**, os passos a serem seguidos p/ a resolução do problema.



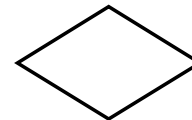
início/fim



saída dados



cálculos



tomada decisão



entrada dados



sentido fluxo

# Fluxograma

- *Desvantagem:* é necessário aprender a simbologia dos fluxogramas, além disso o *algoritmo resultante não apresentará muitos detalhes*, dificultando sua transcrição para um programa.

## Pseudocódigo ou Português Estruturado

- Consiste em analisar o enunciado do problema e escrever, por meio de **regras predefinidas**, os passos a serem seguidos p/ a resolução do problema.
- **Utilizaremos esta abordagem p/ a construção de algoritmos.**

# Pseudocódigo - Exemplo

Faça um algoritmo que mostre a multiplicação de dois números.

## Algoritmo Multiplica;

Var

N1, N2, M: numérico;

Início

**Imprima** "Digite o valor do 1º número: ";

**Leia** N1;

**Imprima** "Digite o valor do 2º número: ";

**Leia** N2;

$M = N1 * N2$ ;

**Imprima** "Multiplicação: ", M;

Fim.



# Conceito de Variável

- Imagine uma situação onde duas pessoas estão conversando sobre a realização uma operação matemática...

## Conceito de Variável (cont.)

- Maria: “Vamos somar dois números.”
- Maria: “E o primeiro número é 5.”
- João guarda o primeiro número na cabeça, ou seja, em sua memória .
- Maria: “O segundo número é 3.”
- João também guarda o segundo nº em sua memória, sem se esquecer do primeiro nº, ou seja, cada nº foi armazenado em posições #’s da memória.

## Conceito de Variável (cont.)

- Maria: “Qual o resultado da soma?”
- João resgata os valores armazenados na memória, realiza a operação (adição) e responde dizendo que o resultado é 8.

# Conclusão

- **Maria** fez com que **João** executasse o *algoritmo da adição de dois números*.
- **Maria** produziu a entrada de 2 valores numéricos.
- **João** *executou o algoritmo* a partir desta *entrada* e produziu uma *saída* com o resultado **8**.

# Tipos de Dados

- Os tipos de dados são:
  - Numérico
    - Inteiro
    - Real
  - Caractere
  - Lógico (*booleano*)

# Tipos de Dados - Numérico

- Os dados numéricos dividem-se em 2 grupos:  
**Inteiros** e **Reais**
- Os números **inteiros** podem ser *positivos* ou *negativos* e **NÃO** possuem a **parte decimal**
- Esse tipo de dado quando armazenado na memória do computador, ocupa 2 bytes, por isso temos  $2^8 \times 2^8 = 2^{16} = 65.536$  possibilidades de representação dos números inteiros. A faixa de valores inteiros possíveis é: -32.768 até +32.767.

# Tipos de Dados - Numérico

- Os números **Reais** podem ser *positivos* ou *negativos* e **possuem** a **parte decimal**
- Esse tipo de dado quando armazenado na memória do computador, ocupa 4 bytes, por isso temos valores que podem variar a partir de  $2.9 \times 10^{-39}$  a  $1,7 \times 10^{38}$

# Tipos de Dados – Lógico

- São dados **Booleanos** podem assumir os valores *verdadeiro* ou *falso*.



# Tipos de Dados - Caractere

- São formados por um único caractere ou por uma cadeia de caracteres.
- Esses caracteres podem ser letras maiúsculas, minúsculas, números (que não podem ser usados p/ cálculos) e os caracteres especiais (&, #, @, ?, +, etc.).
- Exemplo: 'aluno', '1234', '4 + 2', '0.55', 'AED I'

# Identificadores

- São os nomes das *variáveis*. São as posições de memória onde o computador armazenará os dados manipulados pelo programa.
- Regras básicas:
  - Todo identificador começa por uma letra
  - Não são permitidos espaços em branco e caracteres especiais
  - Não é permitida a utilização de palavras reservadas nos identificadores

# Comando de Atribuição

- É um comando que tem por finalidade **atribuir** valores ou operações ao *Identificador*.

- **Representação:**

==



- **Exemplos**

- $A = 3.45$
  - $X2 \leftarrow 11$
  - $\text{Nome} = \text{'Rubens Barrichello'}$
  - $S \leftarrow \text{Falso}$
  - $X = Y + 3$
- $Z = 4$
  - $Z = Z + 2$

# Comando de Atribuição (cont)

**Resumindo...**

**Variável ← expressão**

# Exercícios

- Classifique os resultados das instruções abaixo de acordo com os tipos de dado (Inteiro, Real, Lógico ou Caractere):

1.  $K \leftarrow 1$
2.  $Cor \leftarrow \text{'verde'}$
3.  $Teste \leftarrow \text{falso}$
4.  $Cod \leftarrow 32 - 5 / F$
5.  $Delta \leftarrow B * B - 4 * A * C$
6.  $X \leftarrow X + 1$

## Exercícios (cont)

7. Dia  $\leftarrow$  'segunda'
8. Fator  $\leftarrow$  '5 / 3 + 7.5'
9. Status  $\leftarrow$  verdadeiro
10. Media  $\leftarrow$  Soma / N
11. Total\_Geral  $\leftarrow$  A + B + C + D
12. J  $\leftarrow$  0

# Operadores e Funções

- **Operadores Aritméticos**

- + adição
  - subtração
  - \* multiplicação
  - / divisão
  - \*\* elevado a

- **Operadores Relacionais**

- = igual
  - != diferente
  - > maior que
  - >= maior ou igual
  - < menor que
  - <= menor ou igual

- **Funções Matemáticas**

- Sen(x)                      seno
- Cos(x)                      cosseno
- Arctg(x)                    arco tangente

- **Operadores Lógicos**

E

Ou

Não



# Prioridade entre os Operadores

1º) Parênteses mais internos

2º) \*\*

3º) \*                      /

4º) +                      -

# Prioridade entre todos os Operadores

1º) Parênteses mais internos

2º) Aritmético

3º) Relacional

4º) Não

5º) E

6º) Ou

## Exercícios – Calcule o resultado das instruções abaixo:

1.  $\text{valor} \leftarrow 3 + 2 - 1$
2.  $x \leftarrow (3 - 2) * 5 / 2$
3.  $a \leftarrow (8 + 2 - (20 * 2) + 3) / 9$
4.  $s \leftarrow 40 / 2 * (1 + 1)$
5.  $\text{total} \leftarrow 5 + 3 - 4 / 2$
6.  $x3 \leftarrow 8 - 3 * 4 / 6$
7.  $\text{soma} \leftarrow (2 * 5 + (3 - 7 + (45 / 9)) - 1) * -2$
8.  $b \leftarrow 8 + 10 + 6 + 8 / 4$
9.  $w \leftarrow 2 ** 4 / (6 - 2)$

## Comando de ENTRADA

- O comando de entrada é utilizado para receber dados digitados pelo usuário. Os dados recebidos são armazenados em variáveis.
- Representação: **Leia variável;**
- Exemplo: **Leia S;**
  - *Um valor (conteúdo) digitado pelo usuário será armazenado na variável S.*

## Comando de SAÍDA

- O comando de Saída é utilizado para mostrar dados na tela ou na impressora. Esses dados podem ser conteúdos de variáveis ou mensagens.
- Representação: **Imprimavariável;**  
**Imprima 'mensagem';**
- Exemplo: **Imprima S;**  
**Imprima 'O valor da soma é ', S;**

# Estrutura padrão de um Algoritmo

**Algoritmo** nome\_do\_algoritmo;

**Var**

variáveis: **tipo**;

**Início**

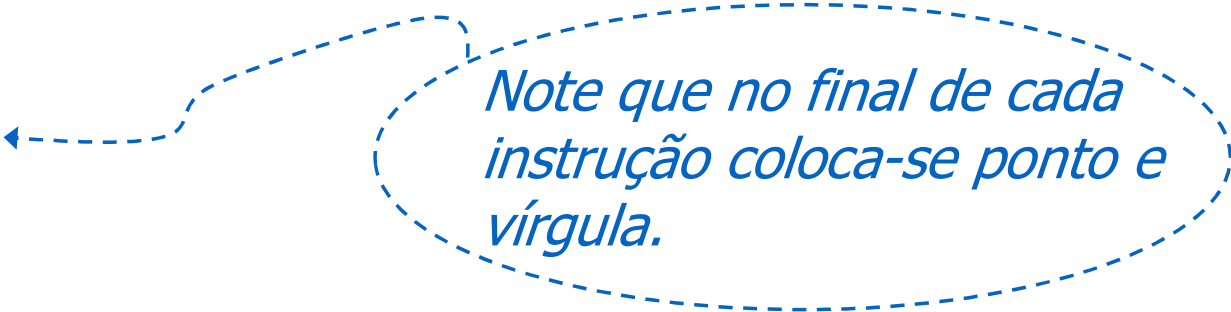
instrução 1;

instrução 2;

...

instrução n;

**Fim.**



*Note que no final de cada instrução coloca-se ponto e vírgula.*

# Exemplo de Algoritmo

- Escreva um algoritmo que leia um valor do tipo Real e, a seguir mostre-o multiplicado por 3.

**Algoritmo** multiplica;

**Var**

a, mult: **real**;

*Tipo das variáveis*

**Início**

**Imprima** “Digite um valor “;

**Leia** a;

mult  $\leftarrow$  a \* 3;

**Imprima** “O valor multiplicado por 3 é “; mult;

**Fim.**

*Imprime o  
conteúdo da  
variável MULT.*



# Exemplo de Algoritmo

- Escreva um algoritmo que leia dois valores do tipo Real e, a seguir mostre a média aritmética entre eles.

**Algoritmo** media\_aritmetica;

**Var**

a, b, media: **real**;

**Início**

**Imprima** “Digite o primeiro valor “;

**Leia** a;

**Imprima** “Digite o segundo valor “;

**Leia** b;

media  $\leftarrow (a + b) / 2$ ;

**Imprima** “A média entre os valores é “, media;

**Fim.**

