

FIAP



TBD

Computacional Thinking using Python
Conceitos Iniciais & Introdução a Lógica de Programação
Prof. Dr. Daniel Trevisan Bravo

* Material adaptado da Profa. Patrícia Angelini

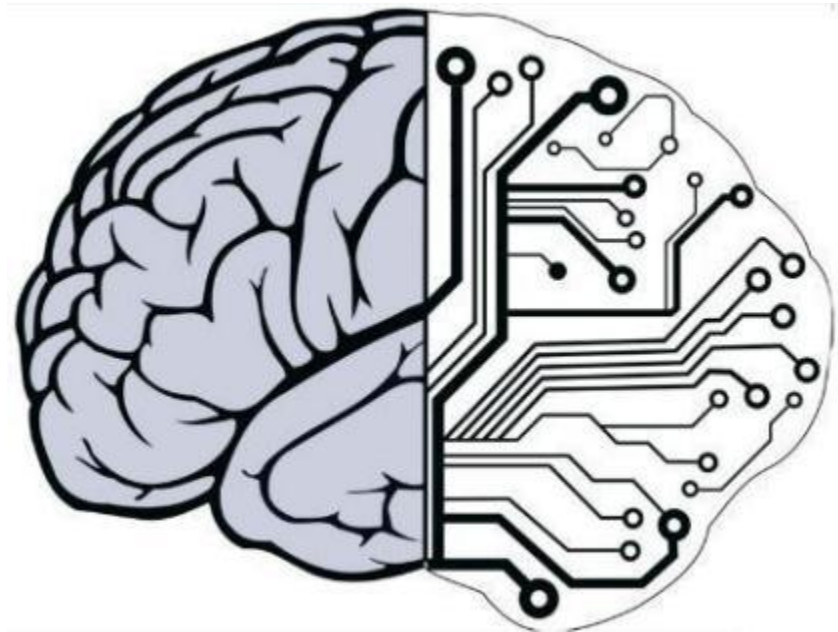
1. CONCEITOS INICIAIS

PENSAR COMO UM COMPUTADOR

pensamento¹

pen.sa.men.to¹

sm (*pensar+mento*²) **1** Ato ou faculdade de pensar. **2** Ato do espírito ou operação da inteligência. **3** Fantasia, imaginação, sonho. **4** Cuidado, preocupação, solicitude. **5** Ideia, lembrança. **6** Modo de pensar; opinião. **7** Alma, espírito. **8** Conceito, moralidade (de um apólogo, epigrama, ou sátira); a intenção de um autor. **P. de classe, Sociol:** conjunto de ideias, valorações, atitudes e conceitos peculiares aos membros de uma camada social. **P. social, Sociol:** conjunto das reflexões não sistemáticas do homem sobre suas experiências como ser social.



COMO ENSINAR O COMPUTADOR A FAZER ALGO?

**TUDO
QUE MEU
COMPUTADOR
SABE
FUI EU QUEM
ENSINEI.**

programae.org.br

COMO O COMPUTADOR REALIZA TAREFAS?

- Ele executa instruções bem definidas passadas por meio de uma **LINGUAGEM DE PROGRAMAÇÃO**
- As pessoas que ensinam os computadores a realizar tarefas são os **PROGRAMADORES.**



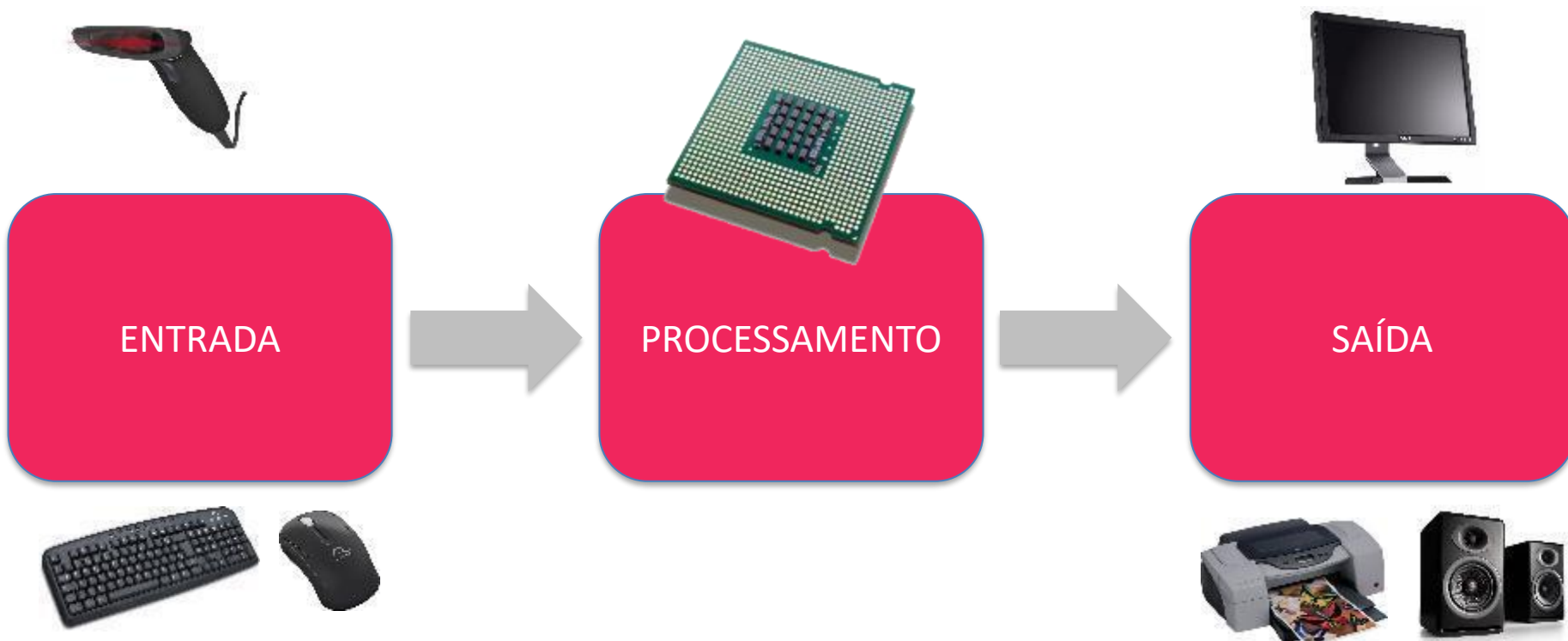
O QUE ESPERAMOS DOS ALUNOS?

**ESPERAMOS QUE
DESENVOLVAM MAIS QUE UMA
HABILIDADE, UM TALENTO.**

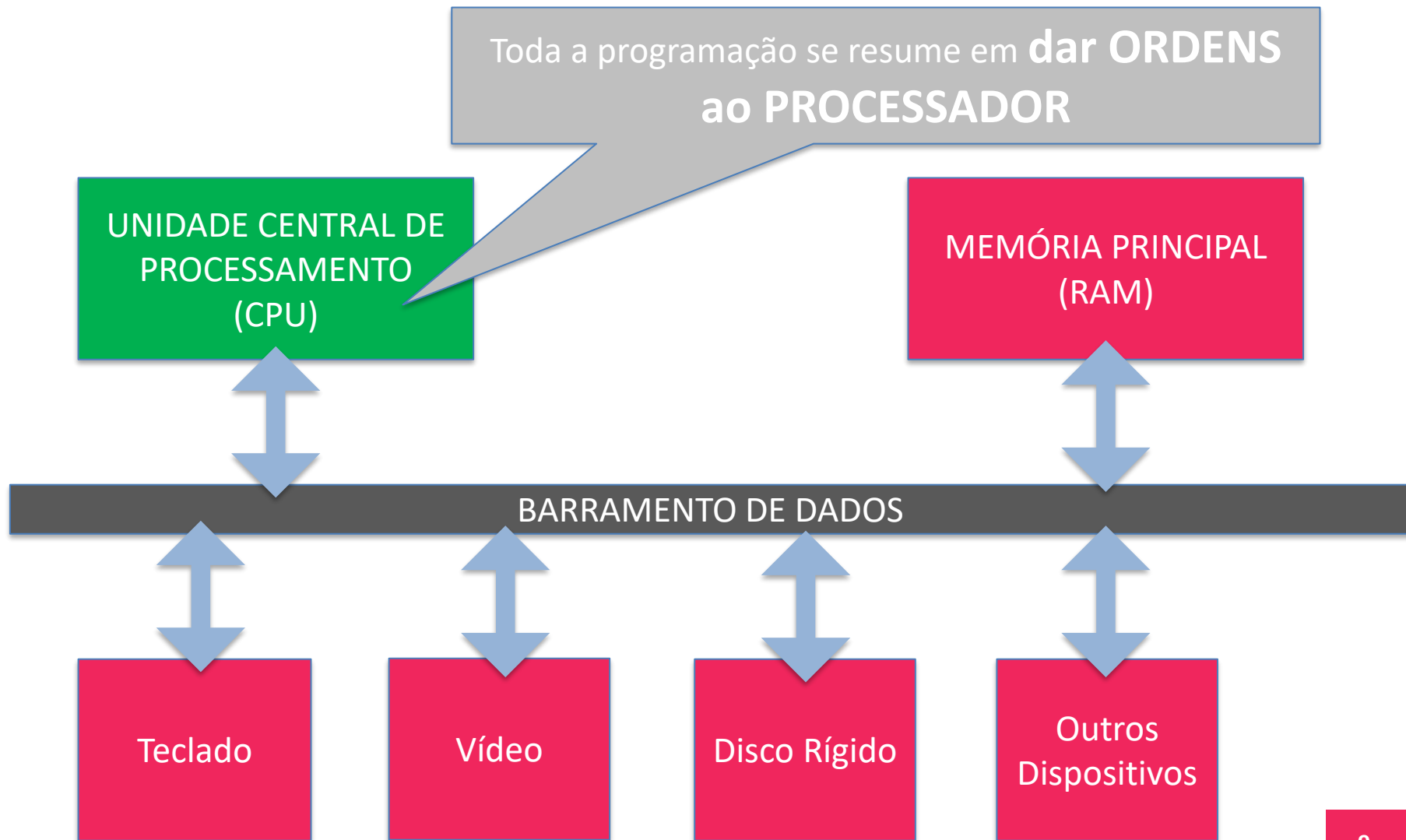
**NÃO ESPERAMOS QUE
DOMINEM UM CONTEÚDO**



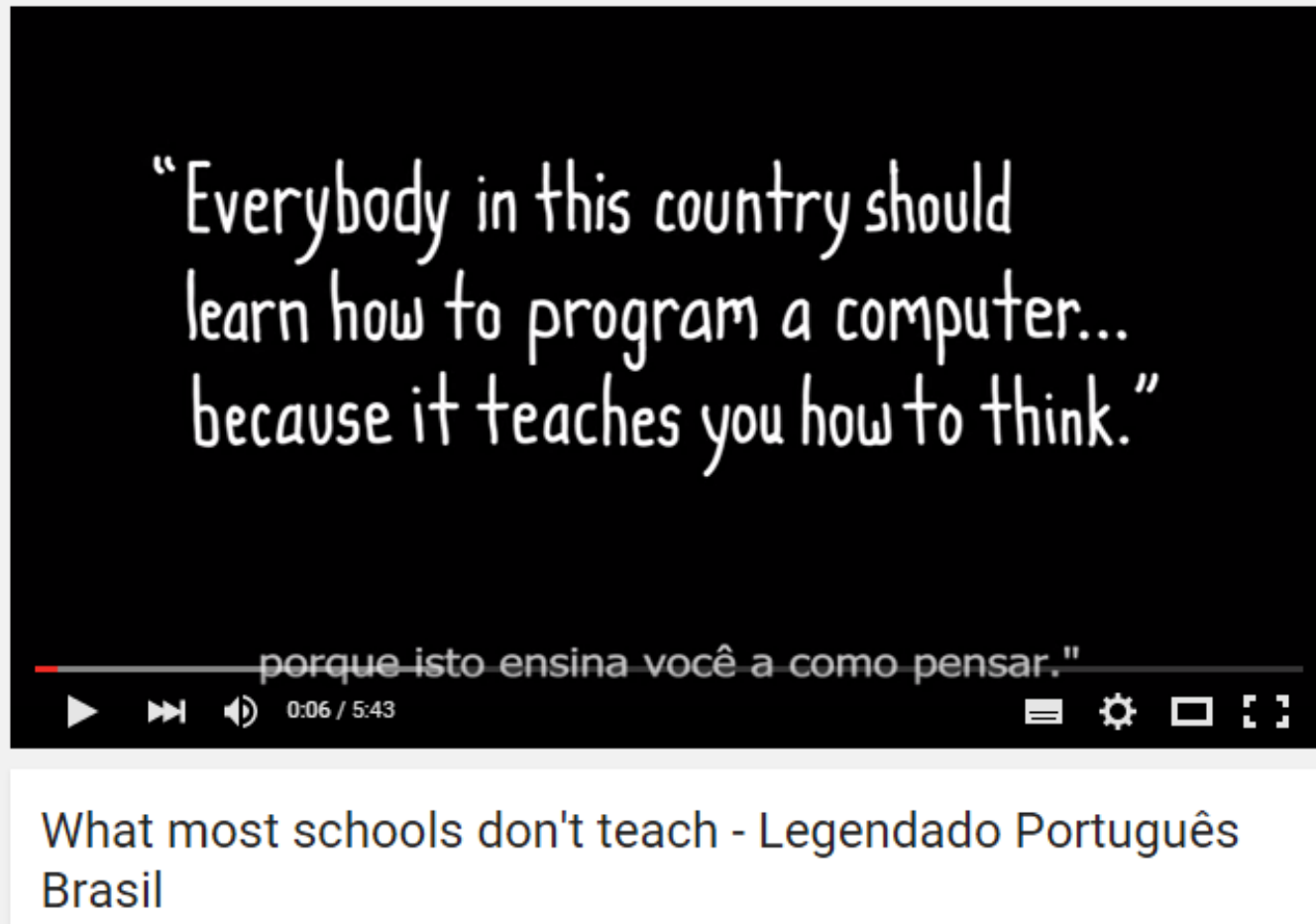
PRINCÍPIO BÁSICO DA INFORMÁTICA



ARQUITETURA DE VON NEUMANN RESUMIDA



UM PEQUENO VÍDEO SOBRE SER PROGRAMADOR



LINK: https://www.youtube.com/watch?v=iKKOV4yGI_M&feature=share

2. INTRODUÇÃO À LÓGICA DE PROGRAMAÇÃO

| LÓGICA

Pode-se entender como lógica, a ciência dos princípios formais do raciocínio.

Todos têm a capacidade de raciocínio, porém é necessário **representá-la formalmente**. A lógica estuda a organização do pensamento ou **raciocínio estruturado**.

- A lógica de programação é a base para o desenvolvimento de todo e qualquer tipo de programa de computador e é representada por meio de **algoritmos**.

ALGORITMOS

Entende-se por algoritmo uma **sequência finita** de instruções que não pode ter duplo sentido e que pode ser executada **mecanicamente**

De forma prática é como uma “receita de bolo” dando ao computador instruções detalhadas sobre como executar determinada tarefa. Essa sequência de regras formais normalmente envolve expressões matemáticas para a resolução de um problema.

Problema é uma **proposta duvidosa** que pode ter múltiplas soluções ou mesmo não ter nenhuma, **logo cada indivíduo pode construir sua própria solução.**

EXEMPLOS DE ALGORITMOS

EXEMPLO 1

Problema das Torres de Hanói: inicialmente têm-se três hastes (A, B e C) e, na haste A, repousam três anéis de diâmetros diferentes, em ordem decrescente por diâmetro.



EXEMPLOS DE ALGORITMOS

OBJETIVO

Transferir os três anéis da haste A para C, usando B se necessário. As regras de movimento são as seguintes:

- deve-se mover um único anel por vez;
- um anel de diâmetro maior nunca pode repousar sobre algum outro de diâmetro menor.



EXEMPLOS DE ALGORITMOS

ALGORITMO

INÍCIO

1. Mover um anel da haste A para a haste C.
2. Mover um anel da haste A para a haste B.
3. Mover um anel da haste C para a haste B.
4. Mover um anel da haste A para a haste C.
5. Mover um anel da haste B para a haste A.
6. Mover um anel da haste B para a haste C.
7. Mover um anel da haste A para a haste C.

FIM

EXEMPLOS DE ALGORITMOS

EXEMPLO 2

Problema dos Missionários e Canibais:

Três missionários e três canibais precisam atravessar um rio. Para tal, dispõem de um barco com capacidade para duas pessoas. Por medida de segurança, não se deve permitir que em alguma margem a quantidade de missionários seja inferior à de canibais

EXEMPLOS DE ALGORITMOS

ALGORITMO

INÍCIO

1. Atravessar um missionário e um canibal para a margem B.
2. Voltar o missionário para a margem A.
3. Atravessar dois canibais para a margem B.
4. Voltar um canibal para a margem A.
5. Atravessar dois missionários para a margem B.
6. Voltar um missionário e um canibal para a margem A.
7. Atravessar dois missionários para a margem B.
8. Voltar um canibal para a margem A.
9. Atravessar dois canibais para a margem B.
10. Voltar um canibal para a margem A.
11. Atravessar dois canibais para a margem B.

FIM

EXERCÍCIO PRÁTICO



Fases para Montar um Algoritmo com solução computacional

- Para montar um algoritmo, é necessário dividir o problema apresentado em três fases fundamentais:



- **Entrada:** dados de entrada
- **Processamento:** procedimentos utilizados para se chegar ao resultado final
- **Saída:** dados processados

EXEMPLOS DE ALGORITMO COM SOLUÇÃO COMPUTACIONAL

Soma de 2 números inteiros definidos pelo usuário



EXERCÍCIOS DE APLICAÇÃO

- Crie um algoritmo que descreva detalhadamente o procedimento para sacar dinheiro em um caixa eletrônico;
- Crie um algoritmo que descreva detalhadamente as etapas necessárias para fritar um ovo;
- Crie um algoritmo que descreva detalhadamente o procedimento para cozinhar um belo miojo.
- Crie um algoritmo que descreva detalhadamente o processo de ir à academia.
- Crie um algoritmo para o problema da Torre de Hanói com 4 anéis.

LEMBRE-SE: Escrever um algoritmo é como ensinar algo a uma criança de 5 anos

TIPOS DE ALGORITMOS

Não estruturados: são aqueles usados para descrever soluções para problemas cotidianos e que não necessariamente podem ser executados por máquinas.

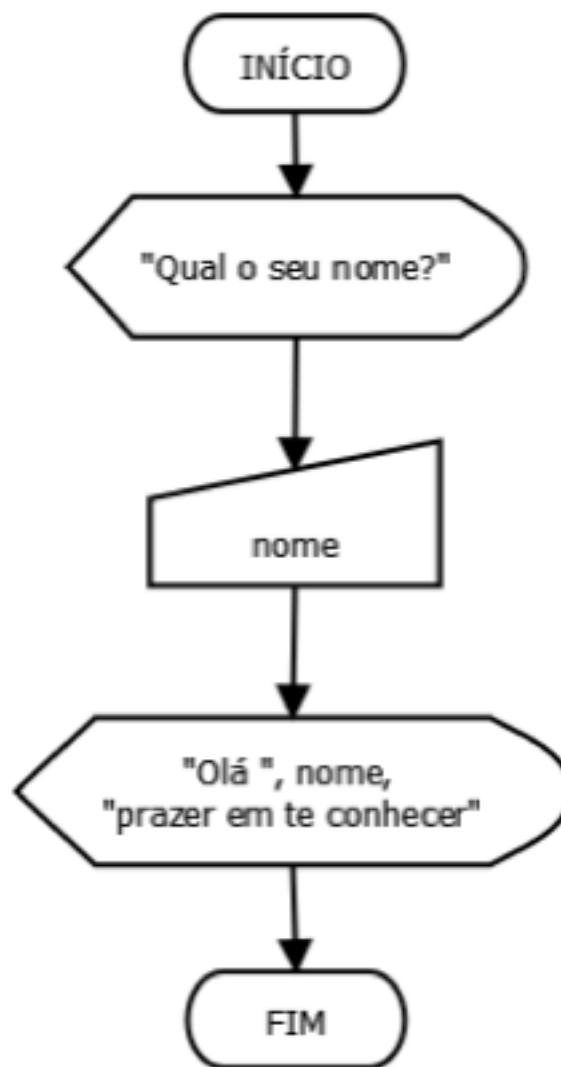
Estruturados: são aqueles usados para descrever instruções detalhadas considerando que as mesmas serão executadas por máquinas.

FORMAS DE REPRESENTAÇÃO DE ALGORITMOS

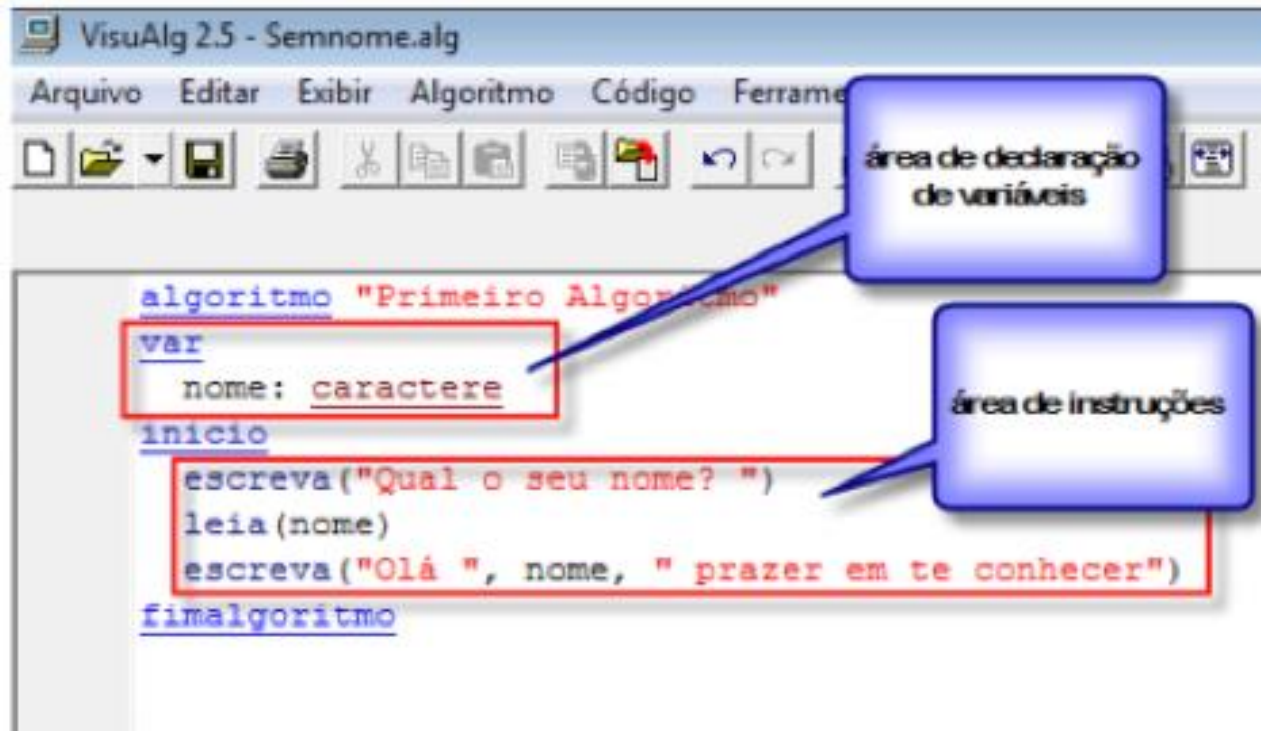
Diagrama de fluxo: também conhecido como diagrama de blocos é uma forma gráfica que representa as instruções de acordo com formas geométricas padronizadas e mundialmente aceitas.

Pseudocódigo: também conhecido como português descreve as instruções em língua portuguesa como se fossem instruções em linguagem nativa.




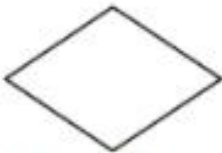
EXEMPLO DE DIAGRAMA DE BLOCOS







EXEMPLO DE PSEUDOCÓDIGO



SIMBOLOS DO DIAGRAMA DE BLOCOS

Símbolo	Função
 TERMINAL	Indica o INÍCIO ou FIM de um processamento Exemplo: Início do algoritmo
 PROCESSAMENTO	Processamento em geral Exemplo: Calculo de dois números
 ENTRA/SAÍDA	Operação de entrada e saída de dados Exemplo: Leitura e Gravação de Arquivos
 DECISÃO	Indica uma decisão a ser tomada Exemplo: Verificação de Sexo

SIMBOLOS DO DIAGRAMA DE BLOCOS

Símbolo	Função
 DESVIO	Permite o desvio para um ponto qualquer do programa
 ENTRADA MANUAL	Indica entrada de dados através do Teclado Exemplo: Digite a nota da prova 1
 EXIBIR	Mostra informações ou resultados Exemplo: Mostre o resultado do calculo
 RELATÓRIO	Relatórios

■ Tipos de Dados

- Os dados são representados por elementos advindos do mundo externo, os quais representam as informações que os seres humanos manipulam. Eles devem ser abstraídos para serem processados por um computador
- São caracterizados por três tipos básicos:
 - Numéricos (inteiros ou reais)
 - Caracteres
 - Lógicos

Inteiros

- Dados numéricos positivos ou negativos
- Exclui qualquer valor numérico fracionário
- Exemplo: 35, 234, -56, -9, 0

■ Reais

- Dados numéricos positivos, negativo e números fracionários
- Exemplo: 35, 234, -56, -9, -45.99, 4.5, 0

■ Caracteres

- Seqüência de valores delimitados por aspas
- Formadas por: letras, números e símbolos
- Também conhecido como: alfanumérico, string, literal, cadeia
- Exemplo: “Programação”, “10a”, “10”, “ ”

■ Lógicos

- Dados com valores que sugerem uma única opção entre duas possibilidades existentes
- Representação: verdadeiro ou falso, 0 (zero) ou 1 (um), sim ou não
- Também conhecido como booleano

■ Variáveis

- Variável é tudo aquilo que é sujeito a variações, que é incerto, instável ou inconstante.
- Como referenciar a armazenar os dados armazenados em um computador?
 - Os valores são armazenados na memória
 - Cada tipo de dado diferente ocupa um número específico de bytes na memória
 - Para recuperar um valor é necessário saber o seu tipo e o endereço do byte inicial ocupado na memória

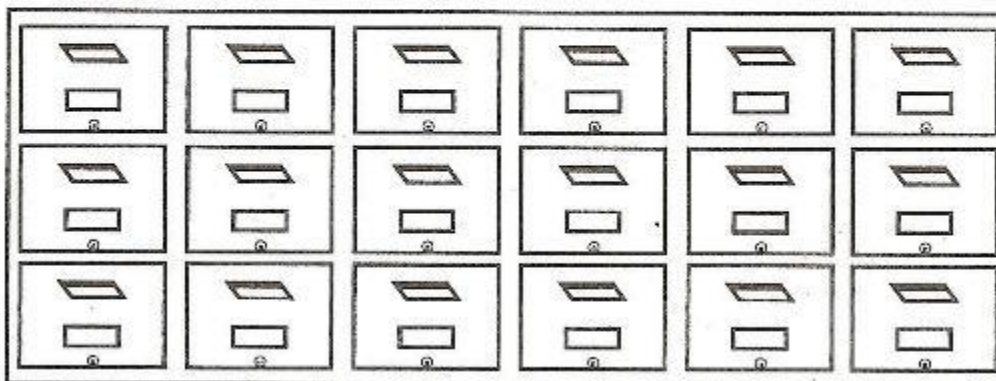
■ Variáveis

- De forma geral, podemos dizer que uma célula de memória está associada a um identificador
- O termo variável é freqüentemente utilizado com sinônimo de identificador
- Variável é uma entidade que guarda valores que podem ser alterados no decorrer de um algoritmo
- Embora uma variável possa assumir diferentes valores, ela só pode armazenar um valor a cada instante

Variáveis

- Uma variável não pode armazenar um valor de tipo de dado diferente daquele para o qual foi criada
- Uma variável deve possuir um nome e este é utilizado para sua identificação e representação dentro de um programa.

Imagine a memória de um computador como um grande arquivo com várias gavetas, e em cada gaveta é possível guardar um único valor por vez. Como em um arquivo, as gavetas devem estar identificadas com uma "etiqueta" contendo um nome.



Variáveis

- Regras de definição e uso de variáveis:
 - O nome de identificação de uma variável pode utilizar um ou mais caracteres
 - O primeiro caractere de identificação do nome de uma variável deve sempre ser alfabético (letras maiúsculas ou minúsculas), os demais podem ser alfanuméricos (letras, números e _)
 - Na definição de um nome composto de uma variável não podem existir espaços em branco entre os nomes
 - Jamais uma variável pode ser definida com o mesmo nome de uma palavra que represente os comandos de uma linguagem de programação de computadores, ou seja, as palavras reservadas de uma linguagem de programação
 - Não pode ser utilizado como nome de variável algum que já tenha sido usado para identificar o nome de um programa

■ Operadores Aritméticos

- Duas categorias: unários e binários
- São unários quando atuam na inversão do estado de um valor numérico
- São binários quando utilizados em operações matemáticas de divisão, multiplicação, adição e subtração
- Em uma expressão aritmética, caso necessite alterar o nível de prioridade de um referido cálculo, ele deve ser definido por meio de parênteses.

Operadores Aritméticos

Operador	Operação	Categoria	Resultado	Prioridade
\leftarrow	Atribuição	-	-	-
+	Manutenção de sinal	Unário	-	1
-	Inversão de sinal	Unário	-	1
\uparrow	Exponenciação	Binário	Inteiro ou Real	2
$\uparrow(1/n)$	Radiciação de n	Binário	Real	2
/	Divisão	Binário	Real	3
*	Multiplicação	Binário	Inteiro ou Real	3
div	Divisão	Binário	Inteiro	3
mod	Módulo (Resto da divisão)	Binário	Inteiro	3
+	Adição	Binário	Inteiro ou Real	4
-	Subtração	Binário	Inteiro ou Real	4

■ Expressões Aritméticas

- São definidas pelo relacionamento existente entre variáveis e constantes numéricas com a utilização dos operadores aritméticos

- Exemplo

`resultado ← 10 * (3 + v1)`

Transformando Expressões Matemáticas em Computacionais

- As expressões aritméticas em computação são escritas de uma forma um pouco diferente da forma conhecida em matemática:
 - Expressão matemática:
$$X = \{43 \cdot [55 : (30 + 2)]\}$$
 - Expressão Computacional:
$$X \leftarrow (43 * (55 / (30 + 2)))$$

Transformando Expressões Matemáticas em Computacionais

- Na forma computacional, as chaves e colchetes são abolidos, utilizando-se em seu lugar apenas parênteses
- É também substituído o sinal de (=) igual pelo sinal de (\leftarrow) implicado ou atribuído
- O sinal implicado ou atribuído (\leftarrow) é utilizado para indicar que o valor de uma expressão aritmética está sendo armazenado em uma variável

Transformando Expressões Matemáticas em Computacionais

- Exemplo: fórmula para calcular a área de um triângulo

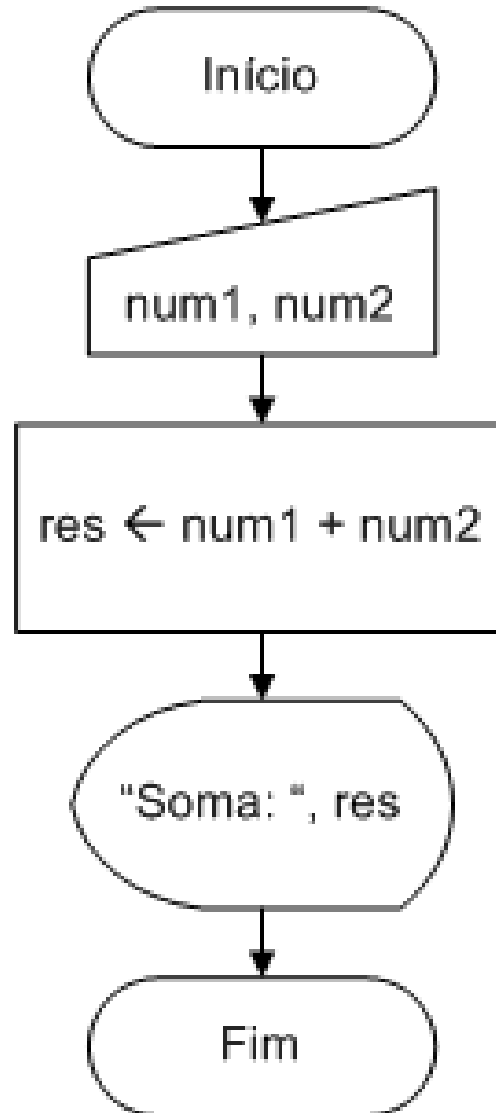
- Expressão matemática:

$$A = \frac{b \cdot h}{2}$$

- Expressão computacional:

$$A \leftarrow (b * h) / 2$$

Diagrama de Blocos – Soma de 2 números inteiros



EXERCÍCIOS DE APLICAÇÃO

Vamos escrever os algoritmos em linguagem agora de maneira estruturada? Use o diagrama de blocos e escreva os algoritmos abaixo.

- Considerando 2 números inteiros, faça a soma, subtração e multiplicação. Por fim, mostre os resultados.
- Calcule e mostre a média aritmética tendo como base 4 avaliações (AV1, AV2, AV3 E AV4).
- Tendo a base e a altura de um triângulo, calcule sua área por meio da fórmula: $\text{area} = (\text{base} \times \text{altura}) / 2$.
- Calcule e mostre o dobro de um número inteiro x.
- Tendo como base o salário de um funcionário, faça o acréscimo de 20% sobre seu valor e exiba-o. DICA: para fazer o acréscimo, multiplique o valor do salário por 1,20.
- Sabendo o peso e a altura de uma pessoa, calcule o IMC (índice de massa corpórea) por meio da fórmula: $\text{IMC} = \text{peso} / (\text{altura} \times \text{altura})$.

#dica: use o LUCID CHART disponível no TEAMS ou o DRAW.IO

REFERÊNCIAS



- OLIVEIRA, Jayr Figueiredo de; MANZANO, José Augusto N. G. **Algoritmos: Lógica para Desenvolvimento de Programação de Computadores**. 23ª Edição. São Paulo: Érica, 2010.
- CONCILIO, Ricardo et al. **Algoritmos e lógica de programação**. 2ª Edição. São Paulo: Cengage, 2011