
Programação de Redes de Computadores

Prof. Gustavo Medeiros de Araújo

gustavo.araujo@sj.ifsc.edu.br

Introdução à Lógica de Programação

- Lógica
 - A lógica de programação é necessária para pessoas que desejam trabalhar com desenvolvimento de sistemas e programas, ela permite definir a sequência lógica para o desenvolvimento.

Então o que é lógica?

Lógica de programação é a técnica de encadear pensamentos para atingir determinado objetivo.

Seqüência Lógica

- Sequência Lógica são passos executados até atingir um objetivo ou solução de um problema.

Instruções

- Na linguagem comum, entende-se por instruções “um conjunto de regras ou normas definidas para a realização ou emprego de algo”.
- Em informática, porém, instrução é a informação que indica a um computador uma ação elementar a executar.

Instruções

- Convém ressaltar que uma ordem isolada não permite realizar o processo completo, para isso é necessário um conjunto de instruções colocadas em ordem sequencial lógica.
- Por exemplo, se quisermos fazer uma omelete de batatas, precisaremos colocar em prática uma série de instruções: descascar as batatas, bater os ovos, fritar as batatas, etc...

Instruções

- **IMPORTANTE:**
- A ordem da execução das tarefas deve ser adequada.
 - não se pode descascar as batatas depois de fritá-las.

Algoritmo

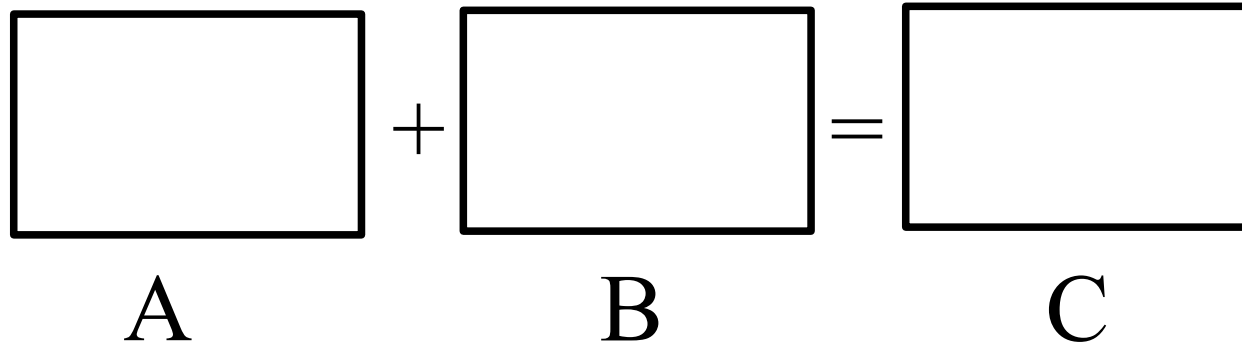
- Um **algoritmo** é formalmente uma **sequência finita de passos** que levam a execução de uma **tarefa**.
- Podemos pensar em **algoritmo** como uma **receita**, uma **sequência de instruções** que dão cabo de uma **meta específica**. Estas **tarefas** não podem ser redundantes nem subjetivas na sua definição, **devem ser claras e precisas**.

Algoritmo

- Exemplo:
 - "Chupar uma bala":
 - Pegar a bala
 - Retirar o papel
 - Chupar a bala
 - Jogar o papel no lixo

Algoritmo

- Somar dois números quaisquer:
 - Escreva o primeiro número no retângulo A
 - Escreva o primeiro número no retângulo B
 - Some o número do retângulo A com número do retângulo B e coloque o resultado no retângulo C:



Programas

- Os programas de computadores nada mais são do que algoritmos escritos numa linguagem de computador
 - (C,C++, Cobol, Java,C#, PHP, etc.)

Exercícios

- Descreva uma sequência lógica para tomar banho.

Desenvolvendo algoritmos

Pseudocódigo:

- alusão à posterior implementação em uma linguagem de programação.
- algoritmos são independentes das linguagens de programação.
- não existe um formalismo rígido de como deve ser escrito o algoritmo.

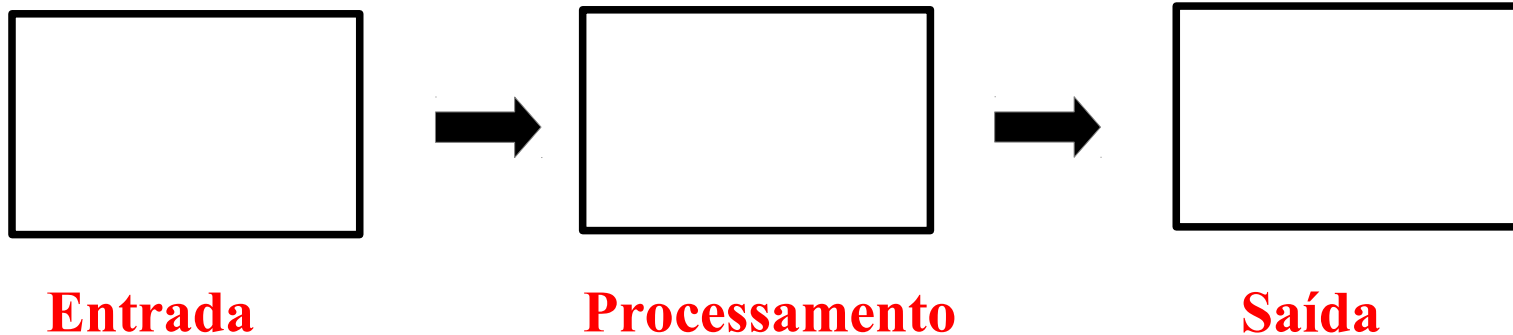
Regras para construção do Algoritmo

Para escrever um **algoritmo** precisamos descrever a **seqüência de instruções**, de maneira **simples e objetiva**. Para isso utilizaremos algumas técnicas:

- Usar somente um verbo por frase
- Imaginar que você está desenvolvendo um algoritmo para pessoas que não trabalham com informática
- Usar frases curtas e simples
- Ser objetivo
- Procurar usar palavras que não tenham sentido dúbio

Fases para Construção de um Algoritmo

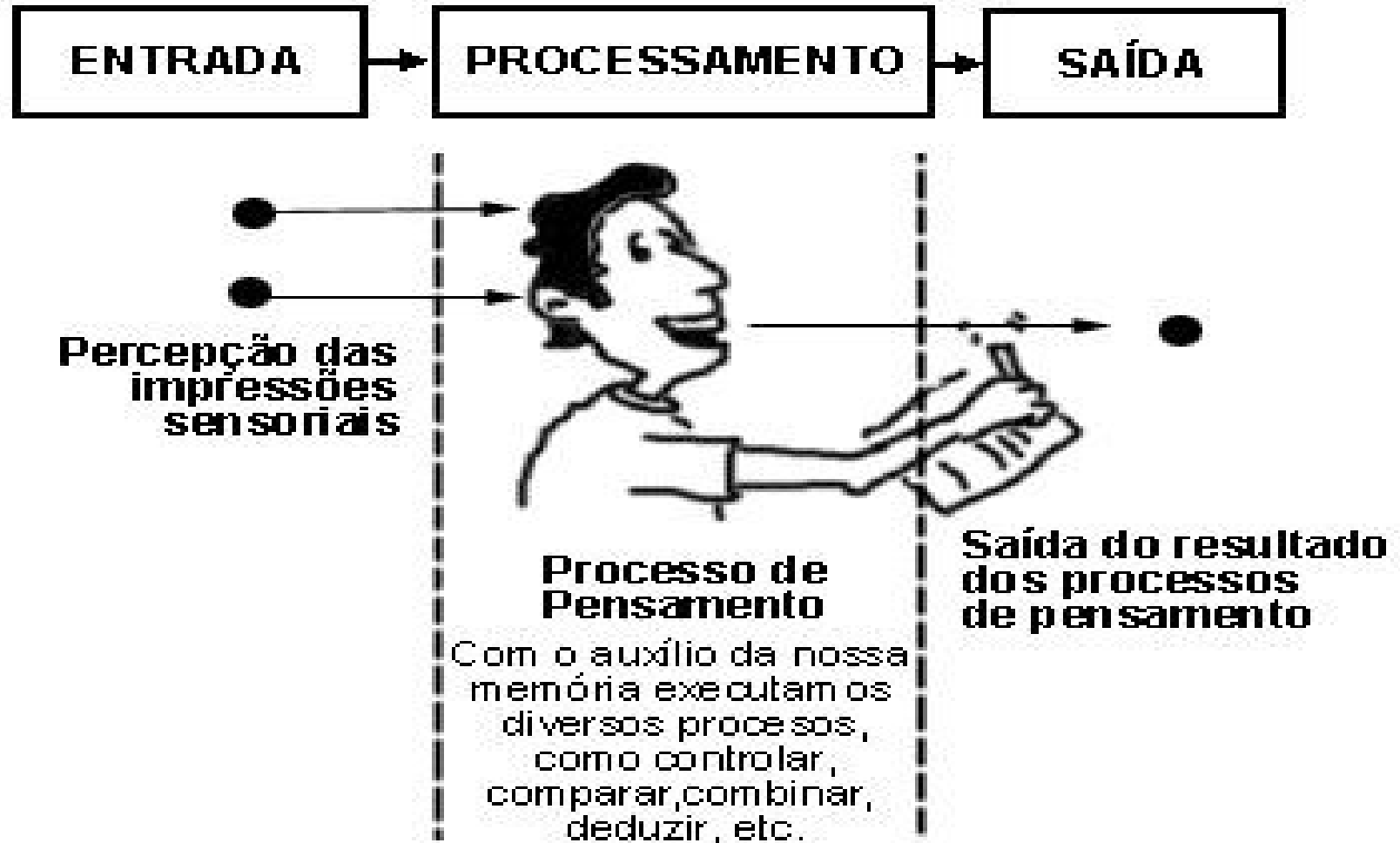
- Para montar um algoritmos é necessário dividir o problemas em:



Fases para Construção de um Algoritmo

- **Entrada:** São os dados de entrada do algoritmo.
- **Processamento:** São os procedimentos utilizados para chegar ao resultado final
- **Saída:** São os dados já processados

Fases para Construção de um Algoritmo



Exemplo de Algoritmo

Imagine o seguinte problema:

- Calcular a média final dos alunos do Curso Técnico. Os alunos realizarão quatro provas: P1, P2, P3 e P4.
- Onde Média final será:
 - $(P1 + P2 + P3 + P4)/4$

Exemplo de Algoritmo

Para montar o Algoritmo proposto, faremos as seguintes perguntas:

- Quais são os dados de entrada?
 - R: Os dados de entrada são P1, P2, P3 e P4
- Qual será o processamento a ser utilizado?
 - R: O procedimento será somar todos os dados de entrada e dividi-los por 4 $((P1 + P2 + P3 + P4)/4)$.
- Quais serão os dados de saída?
 - R: O dado de saída será a média final

Constantes, Variáveis e Tipos de Dados

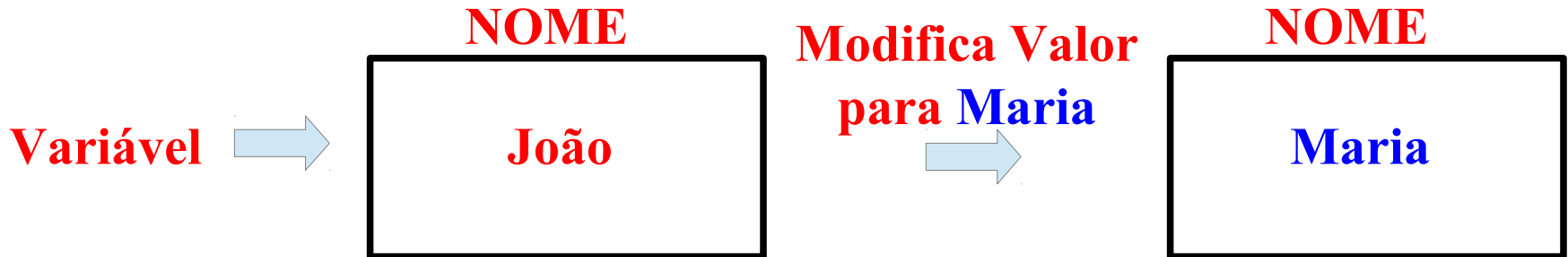
- São os elementos básicos que um programa manipula.
- Uma variável é um espaço reservado na memória do computador para armazenar um tipo de dado determinado.
 - Deve ter nomes para poderem ser referenciadas e modificadas quando necessário.
- Tipos podem ser, por exemplo: inteiros, reais, caracteres, etc. As expressões combinam variáveis e constantes para calcular novos valores.

Constantes

- É um determinado valor fixo que não se modifica ao longo do tempo, durante a execução de um programa.
- **Valor imutável!!**

Variáveis

- Variável é a representação simbólica dos elementos de um certo conjunto. Cada variável corresponde a uma posição de memória, cujo conteúdo pode se alterado ao longo do tempo durante a execução de um programa.



Tipos de Variáveis

- **Numéricas:** Específicas para armazenamento de números, que posteriormente poderão ser utilizados para cálculos.
 - **Inteiras e Reais**
- **Caracteres:** Específicas para armazenamento de conjunto de caracteres que não contenham números (literais). Ex: nomes.
- **Alfanuméricas:** Específicas para dados que contenham letras e/ou números.
- **Lógicas:** Armazenam somente dados lógicos que podem ser Verdadeiro ou Falso.

Tipos de Variáveis

- **Numéricas:** Específicas para armazenamento de números, que posteriormente poderão ser utilizados para cálculos.
 - **Inteiras e Reais**
- **Caracteres:** Específicas para armazenamento de conjunto de caracteres que não contenham números (literais). Ex: nomes.
- **Alfanuméricas:** Específicas para dados que contenham letras e/ou números.
- **Lógicas:** Armazenam somente dados lógicos que podem ser Verdadeiro ou Falso.

Operadores

Os operadores são meios pelo qual incrementamos, decrementamos, comparamos e avaliamos dados dentro do computador. Temos três tipos de operadores:

- Operadores Aritméticos
- Operadores Relacionais
- Operadores Lógicos

Operadores Aritméticos

OPERAÇÃO	SÍMBOLO
Adição	+
Subtração	-
Multiplicação	*
Divisão	/
Exponenciação	**

Operadores Aritméticos

Hierarquia das Operações na seguinte ordem:

- Parênteses ().
- Exponenciação.
- Multiplicação ou Divisão, o que vier primeiro.
- Soma ou Subtração, o que vier primeiro.
- Exemplo:
 - $1 + 7 * 2 ** 2 - 1 = 28$
 - $3 * (1 - 2) + 4 * 2 = 5$

Operadores Aritméticos

Hierarquia das Operações na seguinte ordem:

- Parênteses ().
- Exponenciação.
- Multiplicação ou Divisão, o que vier primeiro.
- Soma ou Subtração, o que vier primeiro.
- Exemplo:
 - $1 + 7 * 2 ** 2 - 1 = 28$
 - $3 * (1 - 2) + 4 * 2 = 5$

Operadores Lógicos

Os operadores lógicos servem para combinar resultados de expressões, retornando se o resultado final é verdadeiro ou falso. Os operadores lógicos são:

- E / AND
- OU / OR
- NÃO / NOT

Operadores Lógicos

Tabela Verdade:

1º Valor	Operador	2º Valor	Resultado
T	AND	T	T
T	AND	F	F
F	AND	T	F
F	AND	F	F
T	OR	T	T
T	OR	F	T
F	OR	T	T
F	OR	F	F
T	NOT		F
F	NOT		T

Operadores Lógicos

Exemplo: Sendo $A = 5$, $B = 8$ e $C = 1$, qual seria o resultados das expressões abaixo?

Expressões			Resultado
$A = B$	AND	$B > C$	Falso
$A <> B$	OR	$B < C$	Verdadeiro
$A > B$	NOT		Verdadeiro
$A < B$	AND	$B > C$	Verdadeiro
$A >= B$	OR	$B = C$	Falso
$A <= B$	NOT		Falso

Estrutura de Decisão e Repetição

Verificamos que na maioria das vezes necessitamos tomar decisões no andamento do algoritmo. Essas decisões interferem diretamente no andamento do programa. Trabalharemos com dois tipos de estrutura. A estrutura de **Decisão** e a estrutura de **Repetição**.

Estrutura de Decisão

SE ENTÃO / IF ... THEN

- Imagine um algoritmo que determinado aluno somente estará aprovado se sua média for maior ou igual a 5.0, veja no exemplo de algoritmo como ficaria:
 - SE MEDIA \geq 5.0 ENTÃO ALUNO APROVADO

Estrutura de Decisão

SE ENTÃO SENÃO / IF ... THEN ... ELSE

- Agora caso a média do aluno seja menor que 5.0 ele será reprovado:

- SE MÉDIA \geq 5.0 ENTÃO
ALUNO APROVADO
SENÃO
ALUNO REPROVADO

-

Estrutura de Decisão

SE ENTÃO SENÃO / IF ... THEN ... ELSE

- Agora caso a média do aluno seja menor que 5.0 ele será reprovado e só será aprovado se a média for maior ou igual a 7.0 e caso esteja entre 5.0 e 7.0, ele terá que fazer outra avaliação:
 - IF MEDIA >= 5 Then
 - IF MEDIA >= 7.0 then
 - Text1 = “Aluno APROVADO”
 - ELSE
 - Text1 = “Aluno Necessita fazer outra Avaliação”
 - ENDIF
 - ELSE
 - Text1 = “Aluno REPROVADO”
 - ENDIF

Comandos de Repetição

- Utilizamos os comandos de repetição quando desejamos que um determinado conjunto de instruções ou comandos sejam executados um número definido ou indefinido de vezes, ou enquanto um determinado estado de coisas prevalecer ou até que seja alcançado.

Comandos de Repetição

Enquanto x, Processar (Do While ... Loop)

Neste caso, o bloco de operações será executado enquanto a condição x for verdadeira.

- Exemplo: Incremente em 1 enquanto a variável A não atingir o valor 100;

A=0

Enquanto A < 100

 A = A + 1;

Fim Enquanto;

- Qual o valor da variável A após a execução?

Comandos de Repetição

Processar ..., Até que x (Do ... Loop Until)

Neste caso, executa-se primeiro o bloco de operações e somente depois é realizado o teste de condição.

- Exemplo: Incremente em 1 enquanto a variável A não atingir o valor 100;

A=0

Faça

 A = A + 1;

Enquanto A < 100;

- Qual o valor da variável A após a execução?

Exercício

1) João Papo-de-Pescador, homem de bem, comprou um microcomputador para controlar o rendimento diário de seu trabalho. Toda vez que ele traz um peso de peixes maior que o estabelecido pelo regulamento de pesca do estado de São Paulo (50 quilos) deve pagar uma multa de R\$ 4,00 por quilo excedente. João precisa que você faça um algoritmo que leia a variável P (peso de peixes) e verifique se há excesso. Se houver, gravar na variável E (Excesso) e na variável M o valor da multa que João deverá pagar. Caso contrário mostrar tais variáveis com o conteúdo ZERO.

- **Dica: para ler um valor use o pseudocódigo: Leia Valor**

Exercício

2) Elabore um algoritmo que leia as variáveis C e N, respectivamente código e número de horas trabalhadas de um operário. E calcule o salário sabendo-se que ele ganha R\$ 10,00 por hora. Quando o número de horas exceder a 50 calcule o excesso de pagamento armazenando-o na variável E, caso contrário zerar tal variável. A hora excedente de trabalho vale R\$ 20,00. No final do processamento imprimir o salário total e o salário excedente.