



Algoritmos e Linguagem de Programação de Computadores II

Ricardo José Cabeça de Souza
www.ricardojcsouza.com.br
ricardo.souza@ifpa.edu.br



SUMÁRIO

- ▶ Resolução de Problemas
- ▶ Pseudo-Linguagem
 - Identificadores
 - Expressões
 - Operadores Aritméticos
 - Operadores Lógicos
 - Comandos Básicos

OBJETIVOS



- ▶ Como resolver problemas
- ▶ Definir a pseudo-linguagem para escrita dos Algoritmos



RESOLUÇÃO DE PROBLEMAS

▶ EXEMPLO

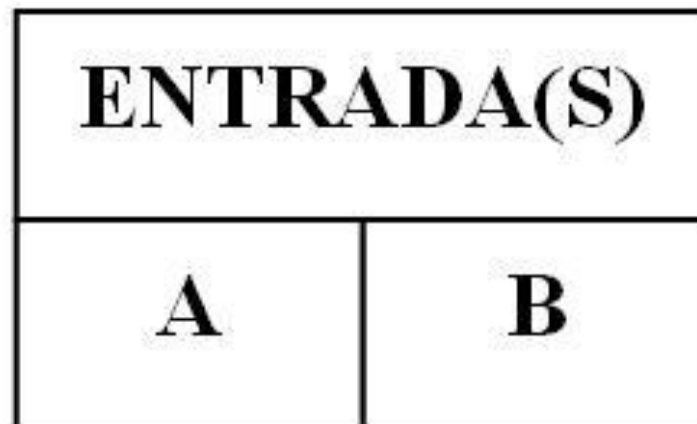
- Problema:
 - Somar dois números
- **DEFINIÇÃO DO PROBLEMA**
 - Somar dois números quaisquer
 - Está claro?
 - Alguma dúvida?



RESOLUÇÃO DE PROBLEMAS

- ▶ **ANÁLISE DO PROBLEMA**
 - ENTRADA

Figura 1.9 – Representação da(s) Entrada(s)





RESOLUÇÃO DE PROBLEMAS

- ▶ **ANÁLISE DO PROBLEMA**
 - COMO RESOLVER?

Figura 1.10 – Representação da Operação

$$\mathbf{A + B}$$



RESOLUÇÃO DE PROBLEMAS

- ▶ **ANÁLISE DO PROBLEMA**
 - COMO RESOLVER?

Figura 1.11 – Resultado da Operação





RESOLUÇÃO DE PROBLEMAS

- ▶ **ANÁLISE DO PROBLEMA**
 - SAÍDA

Figura 1.12 – Representação da Saída





RESOLUÇÃO DE PROBLEMAS

► Seqüência de operações para Somar 2 números:

- a) Entrar com o primeiro valor (A);
- b) Entrar com o segundo valor (B);
- c) Realizar a operação matemática:

$$R \leftarrow A + B$$

- d) Apresentar o resultado da operação (R).



PSEUDO-LINGUAGEM

- ▶ Forma padronizada de realizar a escrita dos algoritmos
- ▶ Padrão de escrita para representar as ações a serem executadas nos nossos algoritmos



PSEUDO-LINGUAGEM

► Identificadores

- Nome usado para designar uma entidade em um algoritmo ou programa
- Qualquer objeto utilizado para representar um elemento no algoritmo



PSEUDO-LINGUAGEM

► Nomeação de Identificadores

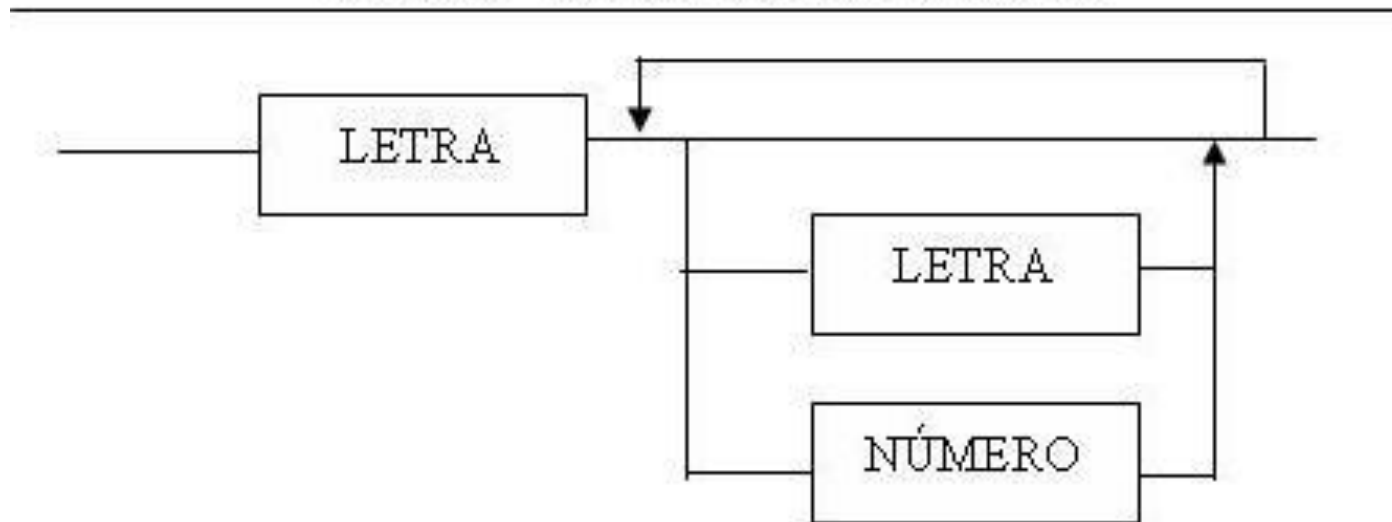
- começar por caractere alfabético (letra)
- podemos utilizar caracteres alfabéticos (letras) e números em sua composição
- defina nomes significativos
- Não utilizar caracteres especiais
- Não utilizar espaços em branco
- Não utilizar palavras reservadas na linguagem de programação



PSEUDO-LINGUAGEM

► Nomeação de Identificadores

Figura 1.13 – Nomeação de Identificadores





PSEUDO-LINGUAGEM

- ▶ **Definição de Tipos para os Identificadores**
 - Constante
 - Variável

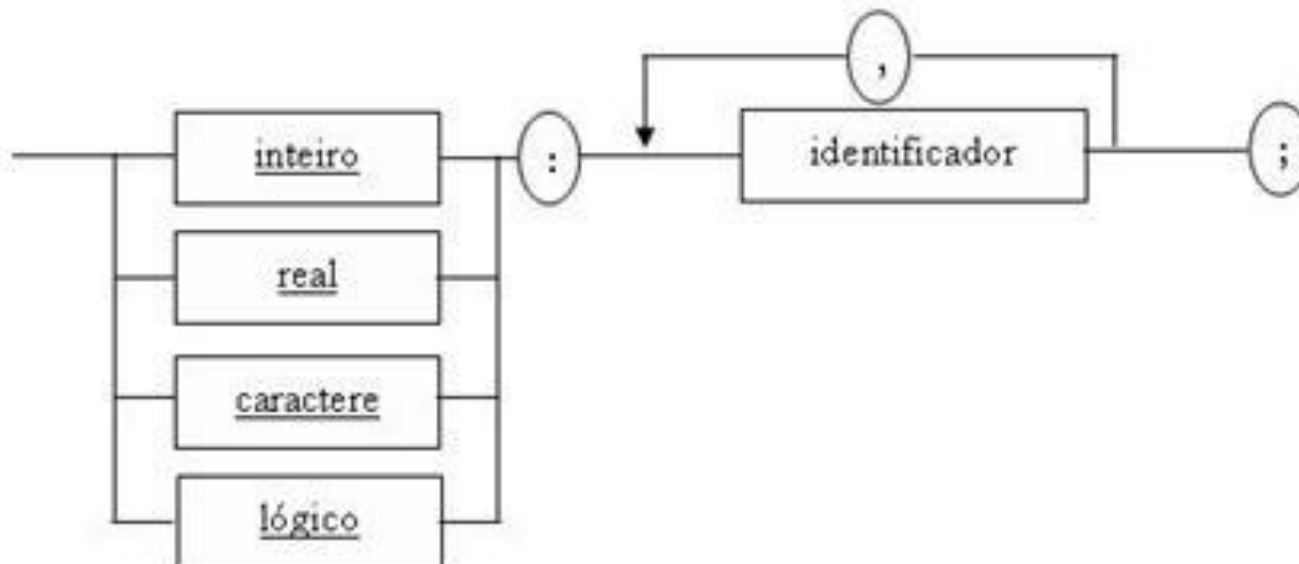
- ▶ **Definição de Tipos de Dados possíveis para armazenamento em Identificadores (variável ou constante)**
 - Inteiro
 - Real
 - Caractere
 - Lógico



PSEUDO-LINGUAGEM

- Declaração de Identificadores (variável ou constante)

Figura 1.16 – Declaração de Identificadores





PSEUDO-LINGUAGEM

► Expressões

- É uma combinação de variáveis, constantes e operadores, e que uma vez avaliada, resulta em um valor



PSEUDO-LINGUAGEM

- ▶ **Operadores Aritméticos**
 - são utilizados para realização de cálculos matemáticos



PSEUDO-LINGUAGEM

► Operadores Aritméticos

Tabela 1 – Operadores Aritméticos

| Operador | Operação | Exemplo de Expressões |
|------------|-----------------|--|
| + | Soma | $5 + 3 \rightarrow 8$ |
| - | Subtração | $8 - 2 \rightarrow 6$ |
| * | Multiplicação | $9 * 2 \rightarrow 18$ |
| / | Divisão real | $5 / 2 \rightarrow 2.5$ |
| <u>div</u> | Divisão inteira | $5 \text{ div } 2 \rightarrow 2$ Observação: Neste caso, se faz a divisão até não ser mais possível realizá-la sem a utilização de recursos adicionais, apresentando como resultado o quociente. |



PSEUDO-LINGUAGEM

► Operadores Aritméticos

| Operador | Operação | Exemplo de Expressões |
|------------|---|--|
| <u>mod</u> | Resto da divisão inteira | <u>5 mod 2</u> → 1 |
| <u>**</u> | Potenciação (<u>Exponenciação</u>) | <u>4 ** 2</u> → 16 Observação: Conforme a regra matemática: <ul style="list-style-type: none">▪ <u>a¹</u> = a▪ <u>a⁰</u> = 1▪ <u>a⁻¹</u> = 1/a▪ <u>a⁻ⁿ</u> = 1/<u>aⁿ</u> |



PSEUDO-LINGUAGEM

► Operadores Aritméticos

| | | |
|---------------|---------------|--|
| <u>RAIZ()</u> | Raiz quadrada | <u>RAIZ(25)</u> → 5 Observação: Somente é possível calcular a raiz quadrada de um número não negativo. |
|---------------|---------------|--|



PSEUDO-LINGUAGEM

- ▶ **Prioridade dos operadores**
 - Ordem de execução das operações a ser seguida
 - Acompanha a regra matemática
 - Executando inicialmente as operações entre parênteses e funções pré-definidas



PSEUDO-LINGUAGEM

- ▶ Expressões Lógicas
 - Aquela cujo resultado é um valor lógico (.verdadeiro. ou .falso.)



PSEUDO-LINGUAGEM

► Expressões Lógicas

Tabela 2 – Operadores Relacionais e Lógicos

| Relações | | |
|----------|------------------|--|
| Operador | Operação | Exemplo de Expressões |
| = | Igual a | $6 = 5 \rightarrow$.falso. $9 = 9 \rightarrow$.verdadeiro. |
| <> | Diferente de | $6 < > 5 \rightarrow$.verdadeiro. $9 < > 9 \rightarrow$.falso. |
| > | Maior que | $6 > 5 \rightarrow$.verdadeiro. $9 > 9 \rightarrow$.falso. |
| < | Menor que | $6 < 5 \rightarrow$.falso. $9 < 9 \rightarrow$.falso. |
| >= | Maior ou igual a | $6 > = 5 \rightarrow$.verdadeiro. $9 > = 9 \rightarrow$.verdadeiro. |
| <= | Menor ou igual a | $6 < = 5 \rightarrow$.falso. $9 < = 9 \rightarrow$.verdadeiro. |



PSEUDO-LINGUAGEM

► Expressões Lógicas

| Lógicos | | |
|------------|-----------|--|
| Operador | Operação | Exemplo de Expressões |
| <u>e</u> | Conjunção | $6 = 5 \text{ e } 9 \leq 9 \rightarrow \text{.falso.}$ |
| <u>ou</u> | Disjunção | $6 = 5 \text{ ou } 9 \leq 9 \rightarrow \text{.verdadeiro.}$ |
| <u>não</u> | Negação | $\text{não}(9 \leq 9) \rightarrow \text{.falso.}$ $\text{não}(6 = 5) \rightarrow \text{.verdadeiro.}$ |



PSEUDO-LINGUAGEM

► Resultado de Operações Lógicas

Tabela 3 – Resultado de Operações Lógicas

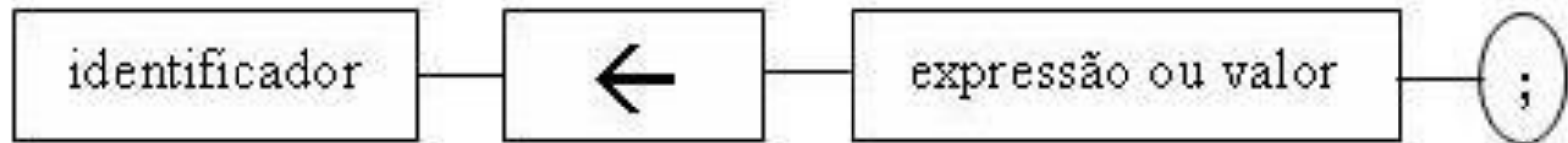
| Proposição 1 | Operador | Proposição 2 | Resultado |
|--------------|-----------|--------------|-----------|
| V | <u>e</u> | V | V |
| V | <u>e</u> | F | F |
| F | <u>e</u> | V | F |
| F | <u>e</u> | F | F |
| V | <u>ou</u> | V | V |
| V | <u>ou</u> | F | V |
| F | <u>ou</u> | V | V |
| F | <u>ou</u> | F | F |
| Operador | | Proposição | Resultado |
| <u>não</u> | | V | F |
| <u>não</u> | | F | V |



COMANDOS BÁSICOS

- ▶ **Comando de Atribuição**
 - se deseja armazenar certo conteúdo em uma variável para posterior manipulação

Figura 1.17 – Comando de Atribuição





COMANDOS BÁSICOS

► Comando de Atribuição

inteiro: A;

real: B;

caractere: LETRA;

caractere: NOME[60];

lógico: TESTE;

A \leftarrow 5;

B \leftarrow 3.6;

LETRA \leftarrow 'm';

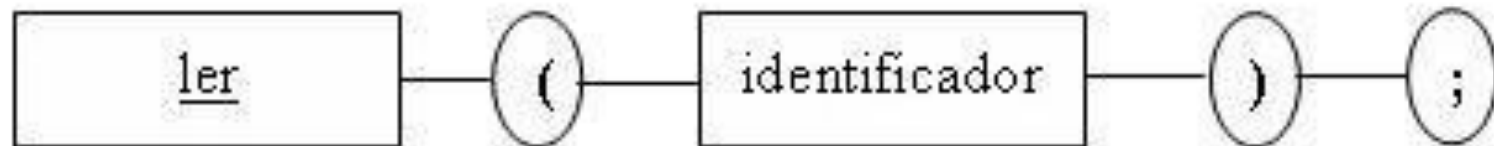
NOME \leftarrow "Governo Federal";



COMANDOS BÁSICOS

- ▶ **Comando de Entrada**
 - são usados para representar a entrada de dados com o uso dos dispositivos de entrada do computador

Figura 1.18 – Comando de Entrada





COMANDOS BÁSICOS

► Comando de Entrada

inteiro: A;

caractere: NOME[60];

ler(A);

ler(NOME);



COMANDOS BÁSICOS

► Comando de Saída

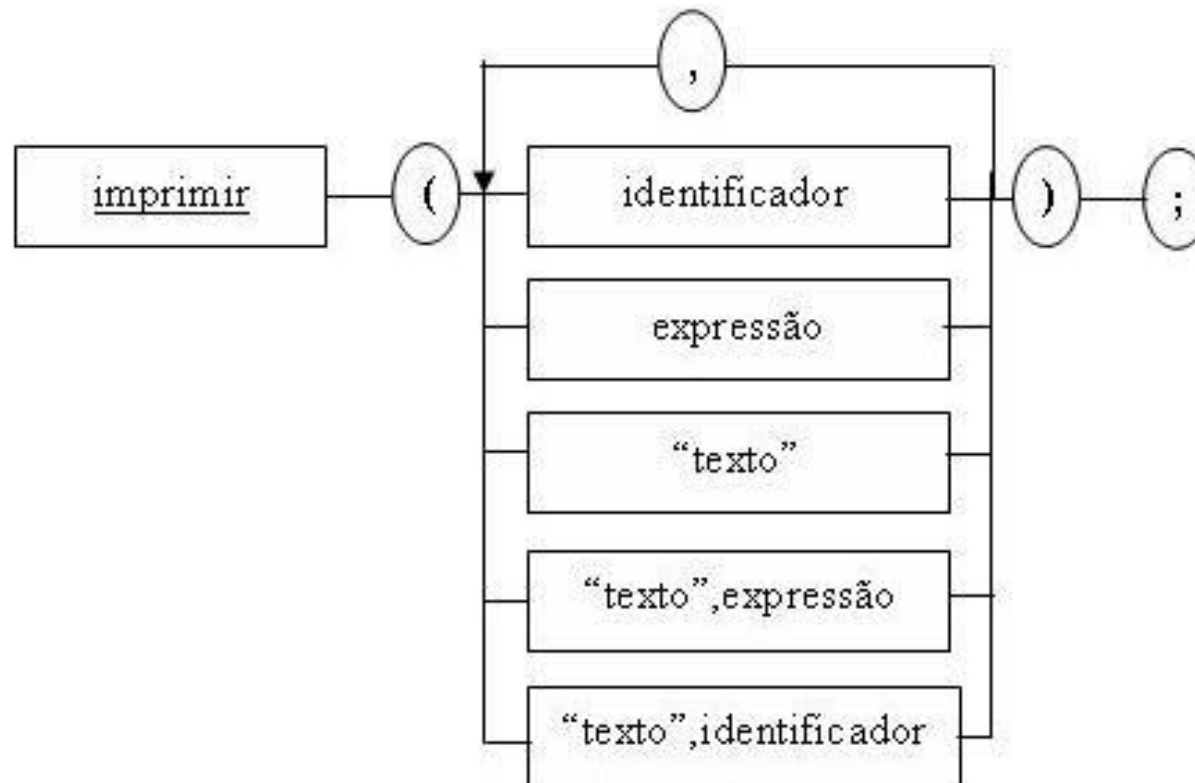
- é utilizado para mostrar o resultado das operações executadas pelo computador, apresentando, conforme definido pelo usuário, a resposta esperada



COMANDOS BÁSICOS

► Comando de Saída


Figura 1.19 – Comando de Saída





COMANDOS BÁSICOS


► Saída com identificador

| Algoritmo | Unidade de Saída |
|--|---|
| <pre>... <u>inteiro</u>: A; A ← 5; <u>imprimir</u>(A); ...</pre> |  |



COMANDOS BÁSICOS

► Saída com expressão

| Algoritmo | Unidade de Saída |
|---|---|
| <pre>... <u>inteiro</u>: A,B; A ← 5; B ← 3; <u>imprimir</u>(A+B); ...</pre> |  |



COMANDOS BÁSICOS

► Saída com “texto”

| Algoritmo | Unidade de Saída |
|--|--|
| <pre>... <u>imprimir</u>("Digite um numero:"); ... <u>imprimir</u>("Boa Noite!"); ...</pre> | <div>Digite um numero: Boa Noite!</div> |



COMANDOS BÁSICOS

- ▶ Saída com “texto” seguido de expressão

| Algoritmo | Unidade de Saída |
|---|--------------------|
| <pre>... <u>inteiro</u>: A,B; A ← 7; B ← 9; <u>imprimir</u>("Soma=",A+B); ...</pre> | <div>Soma=16</div> |



COMANDOS BÁSICOS

- ▶ Saída com “texto” seguido de identificador

| Algoritmo | Unidade de Saída |
|---|--|
| <pre>... inteiro: A,B,RESPOSTA; A ← 7; B ← 9; RESPOSTA ← A + B; imprimir(“Soma entre ”,A, “e”, B, “e igual a ”,RESPOSTA); ...</pre> | <div>Soma entre 7 e 9 e igual a 16</div> |

Referências



- ▶ TANENBAUM, Aaron M. Langsam, Yedidyah, Augenstein, Moshe J. **Estruturas de dados usando C**. São Paulo : MAKRON *Books*, 1995.
- ▶ VELOSO, Paulo. et. al. **Estrutura de dados**. Rio de Janeiro: Campus, 1986.
- ▶ MORAES, Celso Roberto. **Estrutura de dados e algoritmos**. 2. ed. São Paulo: Futura, 2003.
- ▶ CELES, W. Rangel, J. L. **Curso de Estrutura de Dados**. PUC–Rio, 2002.
- ▶ W. Celes, R. Cerqueira, J.L. Rangel. **Introdução a Estruturas de Dados – com técnicas de programação em C**. Rio de Janeiro: Campus, 2004.
- ▶ SCHILDT, Herbert. **C Completo e total**. São Paulo: MAKRON BOOKS, 1997.
- ▶ LOUDON, Kyle. **Dominando algoritmos com C**. São Paulo: CIENCIA MODERNA COMPUTAÇÃO, 2000.
- ▶ JAMSA, Kris. **Programando em C/C++: a bíblia**. São Paulo: Makron Books, 2000.
- ▶ LOPES, Anita. **Introdução a programação: 500 algoritmos resolvidos**. Rio de Janeiro: Campus, 2002.
- ▶ GIMARÃES, Ângelo de Moura. LAGES, Newton Alberto de Castilho. **Algoritmos e estruturas de dados**. Rio de Janeiro: Campus, 1985.
- ▶ LAUREANO, Marcos. **Programando em C para Linux, Unix e Windows**. Rio de Janeiro: BRASPORT LIVROS, 2005.
- ▶ MEDINA, Marco. FERTIG, Cristina. **Algoritmos e programação: teoria e prática**. São Paulo: NOVATEC INFORMATICA, 2005.
- ▶ ARAÚJO, Everton Coimbra de. **Algoritmos: fundamentos e prática**. 2. ed. Ampl. e atual. Florianópolis: VisualBooks, 2005.
- ▶ FEOFILOFF, Paulo. **Projeto de Algoritmos em C**. Disponível em <http://www.ime.usp.br/~pf/algoritmos/aulas/lista.html> acesso em 12/07/2011.
- ▶ HOLANDA, Aurélio Buarque. **Dicionário Aurélio Eletrônico Século XX**. Versão 3.0 Novembro, 1999.