



UNIDADE I

Lógica de Programação e Algoritmos

Profa. Me. Eliane Santiago

Algoritmos

- Algoritmo é qualquer procedimento computacional bem definido, que toma algum valor ou conjunto de valores como entrada e produz um valor ou conjunto de valores como saída.
(CORMEN, 2002)
- Uma sequência finita de instruções aplicadas a um conjunto de dados que permite solucionar classes semelhantes de problemas.
- Um algoritmo é considerado correto quando é aplicável a diferentes instâncias do problema.

Três métodos para escrever algoritmos:

- Descrição Narrativa;
- Fluxograma;
- Pseudocódigo.

Fluxo Processual



Entrada



dados

Processamento



operações

Saídas



resultados

Exemplos de ENTRADA

- Para calcular o volume do cubo é necessário informar o comprimento da aresta.

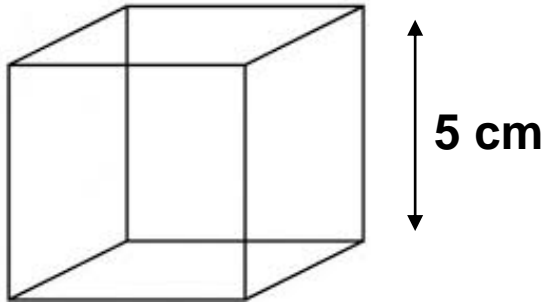
$$V=a^3$$

- Para calcular a equação de 2º grau, é necessário informar as variáveis a, b e c.

$$f(x) = ax^2 + bx + c.$$

Exemplos de ENTRADA para algoritmos

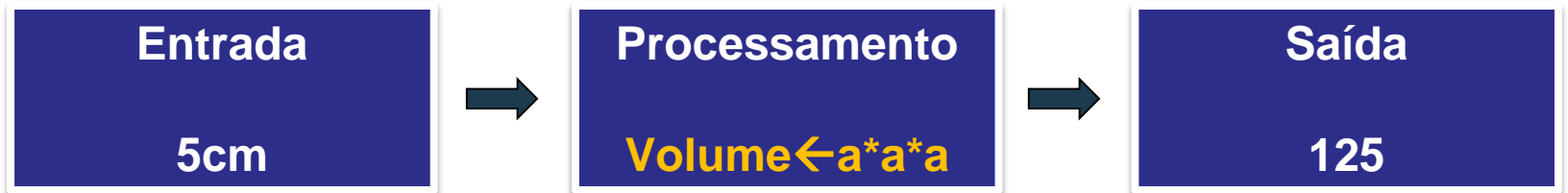
Algoritmo para calcular o volume do cubo



Fonte: acervo pessoal

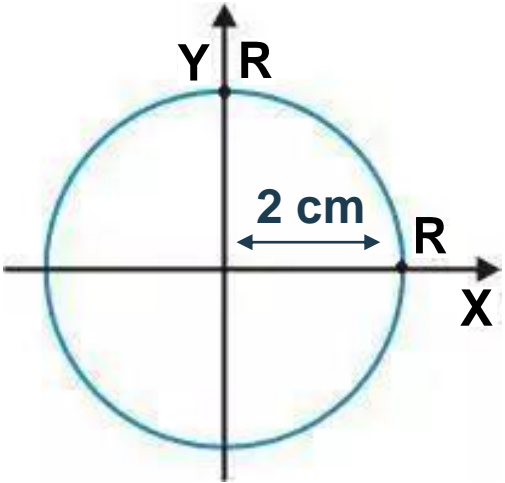
Para calcular o volume do cubo é necessário informar o comprimento da aresta.

Fórmula: $V=a^3$.



Exemplos de ENTRADA para algoritmos

Algoritmo para calcular a área da circunferência



Entrada

raio \leftarrow 2



Processamento

area \leftarrow 3.14*raio*raio



Saída

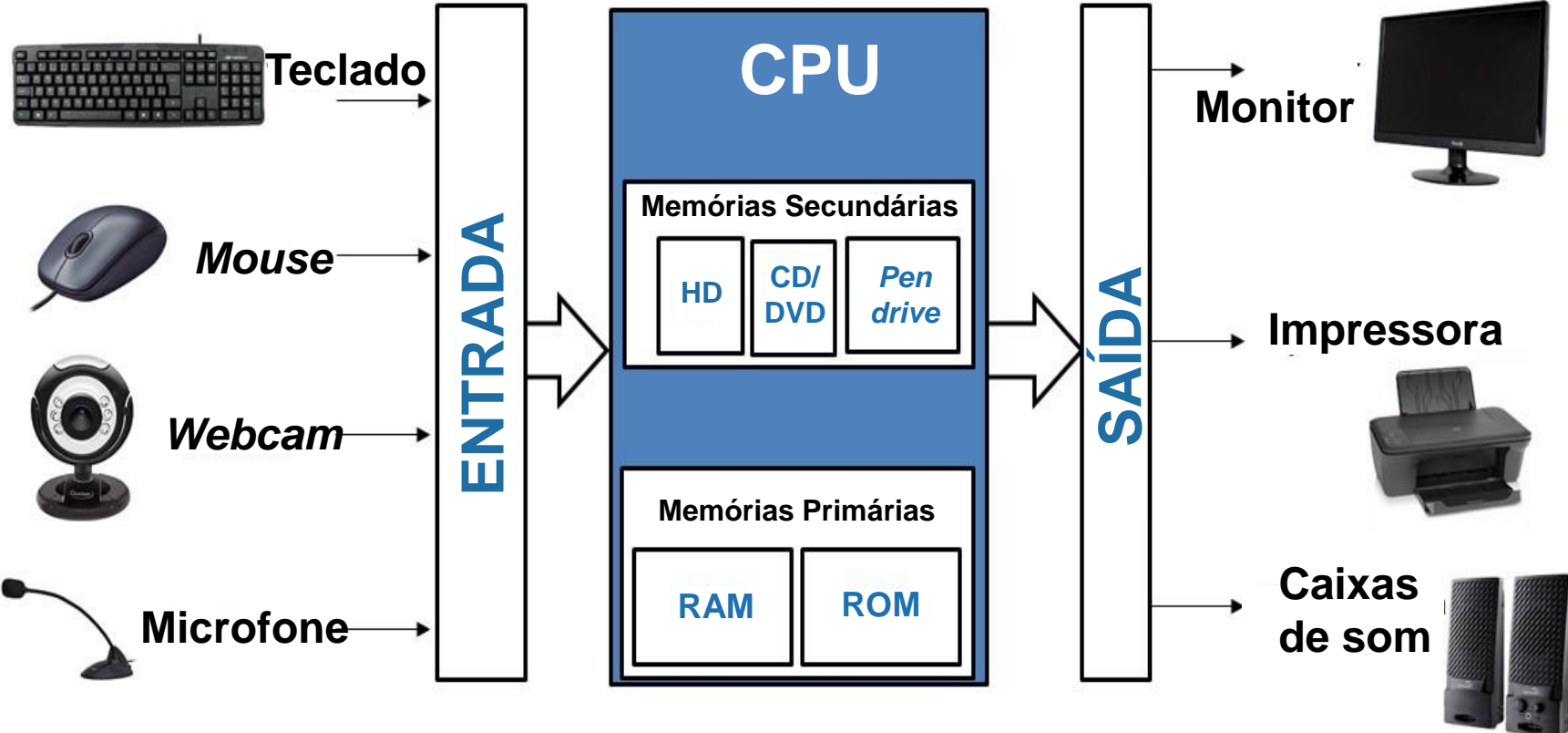
12,56

- Para calcular a área da circunferência, é necessário informar o raio. $A = \pi \cdot r^2$.

Fluxo Processual



Fonte: acervo pessoal



Algoritmo escrito em linguagem natural - Descrição Narrativa

Algoritmo para fazer um Bolo de Liquidificador

1. Separar os ingredientes.
2. Untar a forma.
3. Ligar o forno para pré-aquecer.
4. Bater todos os ingredientes no liquidificador por 5 minutos (exceto o fermento).
5. Adicionar o fermento e mexer delicadamente.
6. Despejar a massa na forma untada.
7. Assar em forno médio (180°) por 40 minutos.



Fonte: <https://pratodoprato.com.br/receita/bolo-de-vo-de-liquidificador-bolo-simples-rapido-e-fofinho-p103>

Fluxo Processual







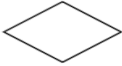



Entradas	↔	lista de ingredientes
Processamento	↔	processar todos os ingredientes no liquidificador.
Saída	↔	bolo

Algoritmo escrito em linguagem gráfica – Fluxograma

- Descrever algoritmos em linguagem gráfica, capaz de representar o processo de solução do problema, comunicar a ideia do processo sem textos massivos.
- No fluxograma **não é necessário expressar a interação com o usuário**, mas sim o fluxo processual de entrada, processamento e saída.

Objetos do Fluxograma ➡

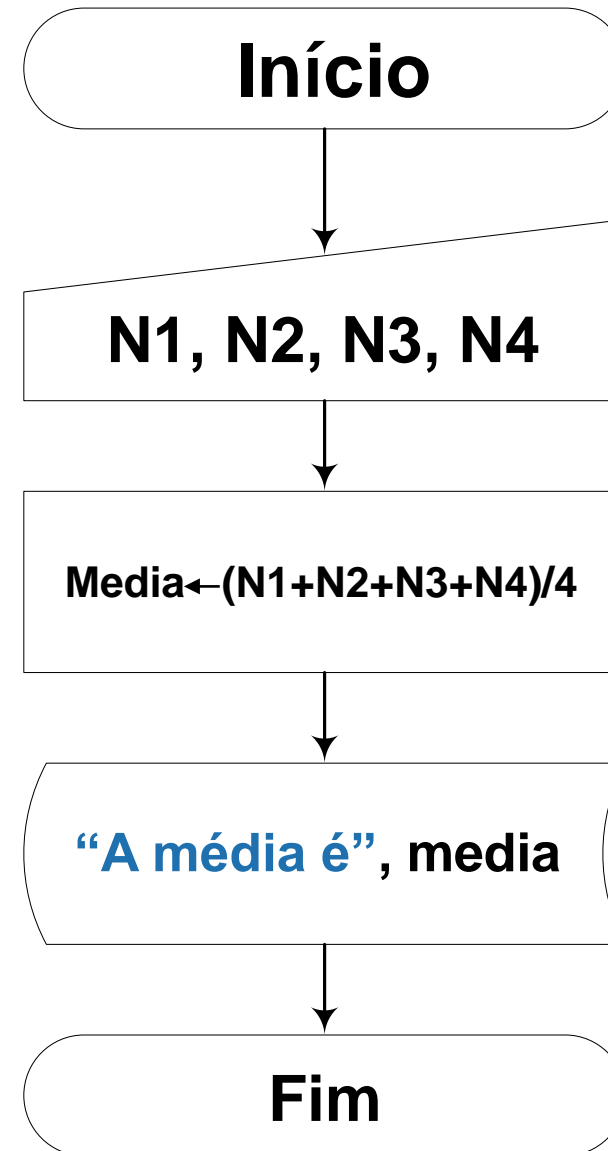
Fonte: acervo pessoal

	Início do Programa
	Entrada de dados
	Processo (comandos, instrução)
	Saída de dados
	Decisão
	Conector
	Fim do Fluxograma
	Módulo ou subprocesso

Algoritmo para calcular e mostrar a média aritmética e exibir o resultado

Descrição narrativa

- *Iniciar o algoritmo Média*
- *Informar 4 notas*
- *Calcular a média aritmética*
- *Escrever o resultado da média*
- *Fim do algoritmo*

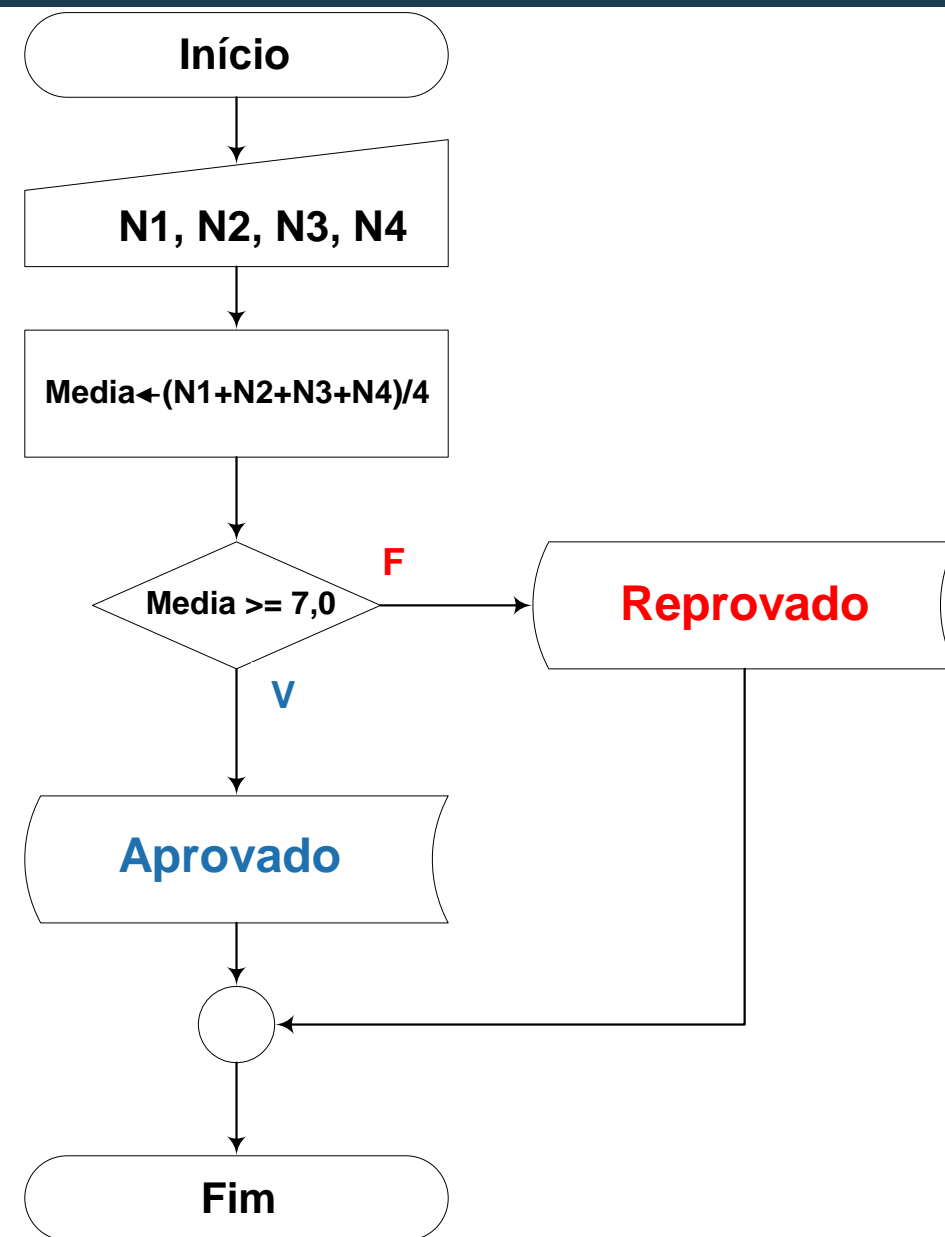


Algoritmo para calcular a média aritmética, a partir da entrada de 4 notas bimestrais, e decidir a situação do aluno

- Regra: se a média for maior ou igual a 7, o aluno é aprovado, senão reprovado.

Descrição narrativa

- *Iniciar o algoritmo Média*
- *Informar 4 notas bimestrais*
- *Calcular a média aritmética*
- **Se** a média for maior ou igual a 7.0, **então** escreva que o aluno está aprovado, **senão** escreva reprovado.
 - *Fim do algoritmo.*



Pseudocódigo

- Uma codificação escrita na estrutura e sintaxe da linguagem de programação, mas com os termos em português.
- Também denominado Portugol.
- A técnica de programação estruturada é um modelo de programação no qual os comandos são executados em ordem sequencial, permitindo desvios por meio de estruturas de decisão ou laços de repetições, bem como pela modularização do código.

Algoritmo <nome_do_algoritmo>

Var

//declaração das variáveis

Início

//sequência de comandos

Fimalgoritmo

Variáveis e Constantes

- **Variáveis** são espaços alocados na memória RAM durante a execução do programa.
- **Constantes** são variáveis cujo valores a ela atribuídos são fixos, ou seja, não podem ser alterados durante a execução do programa.

Variáveis e Constantes podem ser definidas com os tipos

- Inteiro.
- Real.
- Lógico.
- Caracter.

Tipos Primitivos de Dados

Tipos de Dados	Conjunto de valores que aceita
Inteiro	Conjunto dos números inteiros. $Z = \{..., -3, -2, -1, 0, 1, 2, 3, ...\}$
Real	Conjunto dos números reais, ou seja, todos os inteiros e todos os números do intervalo entre dois inteiros.
Lógico	VERDADEIRO ; FALSO
Caractere	letra, número, símbolo ou espaço em branco. Um dado definido como caracter pode receber um número, mas operações matemáticas não poderão ser realizadas sobre ele.

Variáveis: exemplos de declaração

Sintaxes:

`<nome_var> : <tipo>`

`<var1, var2, ..., varN> : <tipo>`

Declarações de variáveis

`base, altura : inteiro`

`salario : real`

`nome: caractere`

`status: lógico`

Constantes: exemplos de declaração

Sintaxe:

`<nome> ← <valor>: <tipo> <const>`

Declarações de constantes

```
MAX ← 100 : inteiro const  
JUROS ← 0.15 : real const
```

Operador de Atribuição (←)

Dados são atribuídos às variáveis considerando o tipo do dado e da variável que o receberá.

Sintaxes:

<var> ← <valor>

<var> ← <var>

<var> ← <expr>

Exemplo:

Algoritmo "Calculo Salario Liquido"

Var

salario_bruto : real

salario_liquido : real

x, y, z : inteiro

Inicio

salario_bruto ← 1798,50

salario_liquido ← salario_bruto * 0,8

x ← 10

y ← x + 10

z ← x + y * 2

Fimalgoritmo

Interatividade

A média anual de uma determinada Instituição de Ensino é calculada de acordo com os fundamentos da média ponderada. Considerando que as notas são expressas em pontos de 0 a 10,0 múltiplos de 0,5 e à cada nota é atribuído pesos 2, 4, 4, a média é dada pela seguinte fórmula:

$$média = \frac{n1 * 2 + n2 * 4 + n3 * 4}{10}$$

Escolha qual opção apresenta as variáveis do problema e os tipos adequados.

Interatividade

- a) N1, N2, N3, media : real ; p1, p2, p3 : inteiro
- b) N1, N2, N3, media, p1, p2, p3 : inteiro
- c) N1, N2, N3, media : real ; p1, p2, p3 : caractere ; media : real
- d) N1, N2, N3 : real ; p1, p2, p3, media : inteiro
- e) N1, N2, N3 : caractere ; p1, p2, p3 : inteiro ; media : inteiro

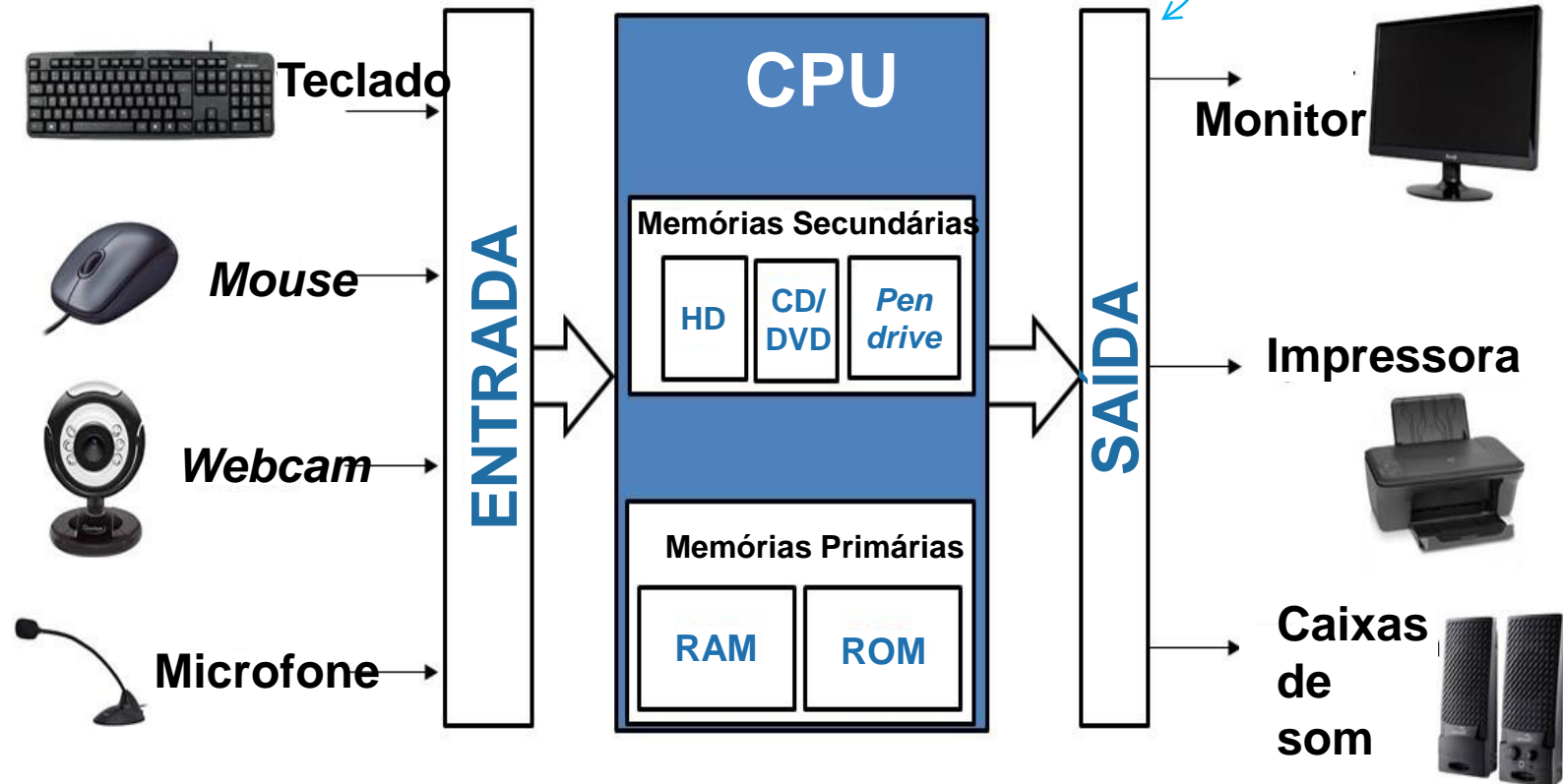
Resposta

- a) N1, N2, N3, media : real ; p1, p2, p3 : inteiro
- b) N1, N2, N3, media, p1, p2, p3 : inteiro
- c) N1, N2, N3, media : real ; p1, p2, p3 : caractere ; media : real
- d) N1, N2, N3 : real ; p1, p2, p3, media : inteiro
- e) N1, N2, N3 : caractere ; p1, p2, p3 : inteiro ; media : inteiro

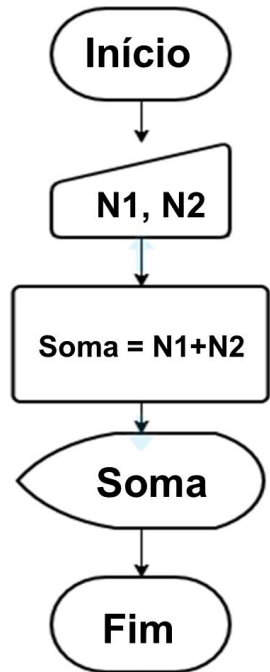
Comandos de Entrada e Saída

- Na programação há comandos específicos para acionar os dispositivos de entrada e saída.
- Comando de entrada: **leia()**
- Comando de saída: **escreva()**

Fonte: acervo pessoal



Algoritmo para somar dois números e exibir o resultado



R A M	n1	15
	n2	30
	soma	45

Algoritmo "Soma"

Var

n1, n2, soma : inteiro

Início

//entrada

escreva("Digite o primeiro número: ")

leia(N1)

escreva("Digite o segundo número: ")

leia(N2)

//processamento

soma = N1 + N2

//saída

escreva("O resultado é : ", soma)

Fimalgoritmo

```
Digite o primeiro número: 15
Digite o segundo número : 30
O resultado é : 45
```

Algoritmo para trocar os valores das variáveis

```
Digite um valor para x:_ 20
Digite um valor para y:_ 10
X = 10 e y = 20
```

Algoritmo TrocaValoresDasVariaveis

Var

x, y, aux : inteiro

Inicio

// entrada

escreva("Digite um valor para x: ")

leia(x)

escreva("Digite um valor para y: ")

leia(y)

// processamento

aux ← x

x ← y

y ← aux

// saída

escreva("x = " , x, " e y = " , y)

Fimalgoritmo

R A M	x	10
	y	20
	aux	20

Operadores

Operadores Aritméticos

- **+** Adição.
- **-** Subtração.
- ***** Multiplicação.
- **/** Divisão.

Operadores Lógicos

- **E**
- **OU**
- **NÃO**

Operadores Relacionais

- **=** Igual
- **<>** Diferente
- **>** Maior
- **>=** Maior ou igual
- **<** Menor
- **<=** Menor ou igual

Operadores Aritméticos e Relacionais

Operadores Aritméticos

$x \leftarrow 5 + 15$

$y \leftarrow (y - 10) * 2$

$z \leftarrow (x + y) / 2$

$r \leftarrow x * x * x$

`escreva (10 * (x + y) / 2 * (z + r))`

`escreva (10 * (x + y) / 2 * (z + r))`

Operadores Aritméticos Auxiliares

- | | | |
|------------------------------------|--------------------------|--|
| ▪ <code>exp(base, expoente)</code> | exponenciação | <code>exp(2, 5)</code>
$2^5 = 32$
<code>x <- exp(5, 3)</code> |
| ▪ <code>raizq(numero)</code> | raiz quadrada | <code>raizq(81) = 9</code> |
| | ▪ <code>div(x, y)</code> | retorna o quociente da divisão de x por y. |
| | ▪ <code>a mod b</code> | retorna o resto da divisão de x por y |

Operadores Relacionais

Operadores Relacionais

- **=** igual
- **<>** Diferente
- **>** Maior
- **<** Menor
- **>=** Maior ou igual
- **<=** Menor ou igual

Operadores Relacionais

Qual é o valor da variável lógica *a* após a seguinte atribuição *a* $\leftarrow 5 * 2 = 4 + 10 / 2$?

$$5 * 2 = 4 + 10 / 2$$

$$10 = 4 + 5$$

$$10 = 9$$

F

Exemplo de algoritmo com operadores relacionais

Algoritmo "Operadores Relacionais"

Var

n1, n2 : inteiro

comparacao : lógico

Início

//entrada

escreval("Número 1: ")

leia(n1)

escreval("Número 2: ")

leia(n2)

//processamento e saída

comparação \leftarrow (n1=n2)

escreval("num1", n1, "é igual ", n2, " ", comparação)

comparação \leftarrow (n1<>n2)

escreval("num1", n1, "é diferente de ", n2, " ", comparação)

comparação \leftarrow (n1>n2)

escreval("num1", n1, "é maior que ", n2, comparação)

//saída

escreval("num1", n1, "é menor que ", n2, n1<n2)

escreval("num1", n1, "é maior ou igual a ", n2, n1=n2)

escreval("num1", n1, "é menor ou igual a ", n2, n1=n2)

Fimalgoritmo

Interatividade

Considere o algoritmo abaixo e responda qual será a saída após a execução. Assuma que as entradas para as variáveis x e y são, respectivamente, 10 e 20.

- a) x=10, y=20, z=20.
- b) x=10, y=20, z=205.
- c) X=20, y=10, z=10.
- d) x=20, y=205, z=20.
- e) x=20, y= 20, z=205.

```
1. Algoritmo "Exercicio01"
2. Var
3.     x, y, z: inteiro
4.
5. Inicio
6.     // entrada
7.     escreval("Digite um valor para x: ")
8.     leia(x)
9.     escreval("Digite um valor para y: ")
10.    leia(y)
11.
12.    // processamento
13.    z <- (x*y) + 5
14.    x <- y
15.
16.    // saída
17.    escreval("x= ", x, " y= ", y, " e z= ", z)
18. Fimalgoritmo
```

Resposta

Considere o algoritmo abaixo e responda qual será a saída após a execução. Assuma que as entradas para as variáveis x e y são, respectivamente, 10 e 20.

- a) x=10, y=20, z=20.
- b) x=10, y=20, z=205.
- c) X=20, y=10, z=10.
- d) x=20, y=205, z=20.
- e) x=20, y= 20, z=205.

```
1. Algoritmo "Exercicio01"  
2. Var  
3.     x, y, z: inteiro  
4.  
5. Inicio  
6.     // entrada  
7.     escreval("Digite um valor para x: ")  
8.     leia(x)  
9.     escreval("Digite um valor para y: ")  
10.    leia(y)  
11.  
12.    // processamento  
13.    z <- (x*y) + 5  
14.    x <- y  
15.  
16.    // saída  
17.    escreval("x= ", x, " y= ", y, " e z= ", z)  
18. Fimalgoritmo
```

Operadores Lógicos

- Os operadores lógicos são expressos pelos símbolos e, ou e não, que significam, respectivamente, a conjunção, a disjunção e a negação.

Operador	Como se lê	Significado
Não	Não lógico	Negação
E	E lógico	Conjunção
Ou	Ou lógico	Disjunção

- Cada sentença afirma (ou nega) um fato, e possui um valor verdade que pode ser VERDADEIRO ou FALSO.

Negação: operador *não* lógico

- A negação é verdadeira se, e somente se a proposição é falsa.
- A proposição é uma sentença declarativa afirmativa ou negativa que pode ser verdadeira ou falsa. A interpretação negar uma verdade, obtém-se uma falsidade e negar uma falsidade, obtém-se uma verdade.

Considere que a proposição:

- **A:** João está aprovado na disciplina LPA.

A	não A
V	F
F	V

- **não A:** João **não está** aprovado na disciplina LPA.

Conjunção: operador e lógico

- A conjunção é verdadeira se, e somente se ambas as proposições são verdadeiras.
- Regra para aprovação na disciplina LPA: Estarão aprovados na disciplina alunos que possuem média maior ou igual a 7,0 **e** frequência igual ou superior a 75% das aulas.

Considere que as proposições A e B signifiquem

- A: média maior ou igual a 7,0.
- B: frequência superior a 75% das aulas.

A	B	A e B
V	V	V
V	F	F
F	V	F
F	F	F

A e B:

- média maior ou igual a 7,0 **e** frequência superior a 75%

Disjunção: operador *ou* lógico

- A disjunção é falsa se, e somente se ambas as proposições são falsas.
- Regra para a habilitar-se a uma vaga de emprego. Procuram-se profissionais de tecnologia com formação em Ciência da Computação **ou** Sistemas de Informação.

Suponha que as proposições A e B signifiquem

- A: formados em Ciência da Computação.
- B: formados em Sistemas de Informação.

A	B	A ou B
V	V	V
V	F	V
F	V	V
F	F	F

A ou B:

- Formados em Ciência da Computação **ou** Sistemas de Informação.

Operadores Lógicos

NAO ! NOT \Leftrightarrow não lógico

E && AND \Leftrightarrow e lógico

OU || OR \Leftrightarrow ou lógico

a) não F

V

não V

F

b) $2 = 10 \bmod 3$ e $-16 < 4$

$2 = 1$ e $-16 < 4$

F

e

V

F

c) $12 < 15$ ou $5 * 2 = 2 + 1$

$12 < 15$ ou $5 * 2 = 2 + 1$

$12 < 15$ ou $10 = 3$

V

ou

F

V

Operadores Aritméticos, Relacionais e Lógicos

Qual é o valor da variável lógica a após a seguinte atribuição?

$a \leftarrow 2 = (2+15) \bmod 3 \text{ OU } -10 > 4$

$$\begin{array}{r} 17 \\ 3 \overline{) 17} \\ \underline{6} \\ 11 \\ \underline{9} \\ 2 \end{array}$$

mod é o resto da divisão

$$\begin{array}{l} 2 = (2+15) \bmod 3 \text{ ou } -10 > 4 \\ 2 = \underline{17 \bmod 3} \text{ ou } -10 > 4 \\ 2 = \underline{2} \text{ ou } \underline{-10 > 4} \\ \underline{\text{V}} \text{ ou } \underline{\text{F}} \\ \text{V} \end{array}$$

O operador negativo multiplica o inteiro 10 por -1, invertendo o sinal

Operadores Lógicos

Qual é o valor da variável lógica a após as seguintes atribuições?

1) $a \leftarrow 2 = 10 \bmod 3 \text{ e } 5 > -7$

2) $12 < 15 \text{ ou } 5 * 2 = 2 + 1$

1) $2 = 10 \bmod 3 \text{ e } 5 > -7$
 $2 = 10 \bmod 3 \text{ e } 5 > -7$
 $2 = 1 \text{ e } 5 > -7$
 $F \text{ e } V$
 F

2) $12 < 15 \text{ ou } 5 * 2 = 2 + 1$
 $12 < 15 \text{ ou } 5 * 2 = 2 + 1$
 $12 < 15 \text{ ou } 10 = 3$
 $V \text{ ou } F$
 V

Exemplo de algoritmo com operadores aritmético, relacional e lógico

- Para ser triângulo, a medida de um lado deve ser menor que a soma dos outros dois.

Algoritmo "Operadores Lógicos"

Var

L1, L2, L3 : inteiro

eh_triangulo : logico

Inicio

//entrada

escreval("Lado 1: ")

leia(L1)

escreval("Lado 2: ")

leia(L2)

escreval("Lado 3: ")

leia(L3)

//processamento e saída

eh_triangulo $\leftarrow ((L1 < (L2 + L3)) \text{ e } (L2 < (L1 + L2)) \text{ e } (L2 < (L1 + L3)))$

//saída

escreval("As medidas formam um triangulo? ", eh_triangulo)

Fimalgoritmo

Interpretações do modelo

- Na conjunção, basta que uma sentença seja FALSA para que a expressão seja falsa.
- Na disjunção, basta que uma sentença seja VERDADEIRA para que a expressão seja verdadeira.

A	B	A e B	A ou B
V	V	V	V
V	F	F	V
F	V	F	V
F	F	F	F

- Cada linha da tabela verdade é uma interpretação do modelo da expressão avaliada.

Interpretações do modelo

Carlos afirma a seus amigos: “**Sábado irei visitar Ana ou Maria**”.

- Se, no sábado, Carlos
 - I. visitar Ana e, também Maria;
 - II. visitar Ana e não visitar Maria;
 - III. não visitar Ana, mas visitar Maria;
 - IV. não visitar nem Ana e nem Maria.

A ou B			
A: Carlos visitou Ana			
B: Carlos visitou Maria.			
A	B	A e B	A ou B
V	V	V	V
V	F	F	V
F	V	F	V
F	F	F	F

Em qual das situações Carlos terá cumprido sua promessa?

- a) I, II e III.
- b) Somente I e II.
- c) Somente I.
- d) Somente II.
- e) Em nenhuma.

Interpretações do modelo

Carlos afirma a seus amigos: “**Sábado irei visitar Ana ou Maria**”.

- Se, no sábado, Carlos
 - I. visitar Ana e, também Maria;
 - II. visitar Ana e não visitar Maria;
 - III. não visitar Ana, mas visitar Maria;
 - IV. não visitar nem Ana e nem Maria.

A ou B			
A: Carlos visitou Ana			
B: Carlos visitou Maria.			
A	B	A e B	A ou B
V	V	V	V
V	F	F	V
F	V	F	V
F	F	F	F

Em qual das situações Carlos terá cumprido sua promessa?

- a) I, II e III.
- b) Somente I e II.
- c) Somente I.
- d) Somente II.
- e) Em nenhuma.

Interatividade

- Analise a expressão lógica e escolha a afirmativa correta.

não F e $\text{raizq}(\text{exp}(7-4,2)) < \text{exp}(5,2)$ ou $15 \bmod (\text{raizq}(49)) \geq 81/9$

**Interprete a expressão em três partes, na forma (A e B ou C), sendo A: Não F;
B: $(\text{raizq}(\text{exp}(7-4,2)) < \text{exp}(5,2))$; e C: $15 \bmod (\text{raizq}(49)) \geq 81/9$.**

- a) A expressão é VERDADEIRA porque A sempre será verdadeiro e o resultado da conjunção (A e B ou C) é verdadeiro sempre que um dos termos é verdadeiro.
- b) A expressão é VERDADEIRA porque C é verdadeiro e na expressão (A e B ou C), basta que C seja verdadeiro para que a expressão seja verdadeira.
 - c) A expressão é VERDADEIRA porque A e B são verdadeiros e na disjunção (A e B ou C), um termo é suficiente para satisfazer a expressão lógica.
 - d) A expressão é FALSA porque a expressão C é falsa e na disjunção, todos os termos devem ser verdadeiros para que a expressão lógica seja verdadeira.
 - e) A expressão é FALSA porque as expressões A, B e C são falsas.

Resposta

não F e **raizq(exp(7-4,2)) < exp(5,2)** ou **15 mod (raizq(49)) >= 81/9**

não F e raizq(exp(**7-4**,2)) < **exp(5,2)** ou 15 mod (**raizq(49)**) >= **81/9**

não F e raizq(exp(**3**,2)) < **25** ou 15 mod **7** >= **9**

não F e raizq(**9**) < **25** ou 1 >= **9**

não F e **3** < **25** ou 1 >= **9**

não F e **V** ou **F**

V e **V** ou **F**

V ou **F**

V

- A expressão é VERDADEIRA porque A e B são verdadeiros e na disjunção (A e B ou C), um termo é suficiente para satisfazer a expressão lógica.

Resposta

- Analise a expressão lógica e escolha a afirmativa correta.

não F e $\text{raizq}(\text{exp}(7-4,2)) < \text{exp}(5,2)$ ou $15 \bmod (\text{raizq}(49)) \geq 81/9$

**Interprete a expressão em três partes, na forma (A e B ou C), sendo A: Não F;
B: $(\text{raizq}(\text{exp}(7-4,2)) < \text{exp}(5,2))$; e C: $15 \bmod (\text{raizq}(49)) \geq 81/9$.**

- a) A expressão é VERDADEIRA porque A sempre será verdadeiro e o resultado da conjunção (A e B ou C) é verdadeiro sempre que um dos termos é verdadeiro.
- b) A expressão é VERDADEIRA porque C é verdadeiro e na expressão (A e B ou C), basta que C seja verdadeiro para que a expressão seja verdadeira.
- c) A expressão é VERDADEIRA porque A e B são verdadeiros e na disjunção (A e B ou C), um termo é suficiente para satisfazer a expressão lógica.
- d) A expressão é FALSA porque a expressão C é falsa e na disjunção, todos os termos devem ser verdadeiros para que a expressão lógica seja verdadeira.
- e) A expressão é FALSA porque as expressões A, B e C são falsas.

Boas Práticas de Programação

- Organizar o programa em blocos.
- Indentar o código.
- Identificar variáveis com nomes significativos.
- Comentar o código para explicar uma lógica.
- Identificar constantes com letras maiúsculas e variáveis com letras minúsculas.

Blocos de programação

- Usam delimitadores de início e fim.
- Ajudam na organização do código.
- Facilitam a compreensão do código.

Bloco Principal:

```
Início  
    comando1  
    comando2  
    :  
    comandoN  
Fimalgoritmo
```

Bloco Se:

```
se (condição) entao  
    comando1  
    comando2  
    :  
    comandoN  
senão  
    comando1  
    comandoN  
fimse
```

Bloco do laço Enquanto:

```
enquanto (condição)  
    comando1  
    comando2  
    :  
    comandoN  
fimenquanto
```

Bloco Faça:

```
faca  
    comando1  
    comando2  
    :  
    comandoN  
ate (condicao)
```

Indentação

- É o deslocamento dentro do bloco.
- Ajuda a identificar a sequência de comandos dentro do bloco.
- Ajuda na visualização de onde inicia e termina o bloco.

- É um recurso da lógica da programação.

Algoritmo "Concatenar Nome e Sobrenome"

Var

nome, sobrenome, nome_completo : caractere

Inicio

//entrada

escreva("Nome: ")

leia(nome)

escreva("Sobrenome: ")

leia(sobrenome)

//processamento

nome_completo <- nome + " " + sobrenome

//saida

escreva("Nome completo : ", nome_completo)

Fimalgoritmo

Exemplo de algoritmo

```
Algoritmo "Equação de primeiro grau"
Var
    x, y, a, b: inteiro
Início
    //entrada
    escreva ("Digite o valor de A.: ")
    leia (a)
    escreva ("Digite o valor de B.: ")
    leia (b)
    escreva ("Digite o valor de X.: ")
    leia (x)

    //processamento
    y <- a*x+b

    //saida
    escreva ( "O valor de y é:", y)
Fimalgoritmo
```

Comentários

- Comentários no código são textos que auxiliam na compreensão da lógica, escrito em linguagem natural.

Blocos de comentários:

/* delimita o início do bloco
***/ delimita o fim do bloco**

```
/*  
* Programa.....: Calcular área do Retângulo  
* Autor(a).....: Eliane Oliveira Santiago  
* Data de criação...: 26/01/2021  
* Última alteração..: 05/02/2021  
* Responsável.....: a autora  
*/
```

Comentários de uma única linha:

```
// entrada  
// processamento  
// saída
```

Algoritmos sequenciais

- Escreva um algoritmo que calcule o volume de um paralelepípedo. Esse algoritmo deve receber três valores reais, um para comprimento, um para largura e um para altura, depois deve apresentar o resultado calculado.

```
Algoritmo "Volume do paralelepípedo"  
Var  
    comprimento, largura, altura : real  
  
Início  
    //entrada  
    escreva("Comprimento do paralelepípedo... (C): ")  
    leia(comprimento)  
  
    escreva("Largura do paralelepípedo..... (L): ")  
    leia(largura)  
  
    escreva("Altura do paralelepípedo..... (A): ")  
    leia(altura)  
  
    //processamento e saída  
    escreva("O Volume do Paralelepípedo é: ", comprimento * largura * altura)  
Fimalgoritmo
```

Método para desenvolver um bom algoritmo

- Ler e compreender o problema para o qual será construído um algoritmo;
- Determinar quais serão a entrada de dados do seu algoritmo;
- Determinar quais as ações, lógicas e/ou aritméticas, que deverão ser realizadas no seu algoritmo, bem como, as restrições, se houver, para cada uma dessas ações;
- Determinar qual será a saída de resultados do seu algoritmo;
- Construir o algoritmo em um dos tipos de algoritmo descritos na próxima seção;
- Testar o algoritmo.

A lógica de Programação

- A estratégia para resolver os problemas.
- O processo de resolução de problemas.
- A sequência correta para executar os comandos.
- A identificação das variáveis e os tipos corretos.

Interatividade

- Dado o algoritmo abaixo, escolha a opção que representa a saída correta.

Algoritmo "Exercicio4"

Var

```
num : inteiro  
falsoverdadeiro : logico  
nome, endereco : caractere  
num_pi : real const
```

Inicio

```
num_pi <- 3.1415  
num <- 10  
falsoverdadeiro <- ((num_pi<3.0) ou (num>=(18 mod 5)))  
nome <- " Fulano "  
endereco <- "Av. Torres de Oliveira, 330"
```

```
escreval(num, nome, endereco)
```

```
escreval(falsoverdadeiro)
```

```
escreval(num_pi)
```

Fimalgoritmo

Interatividade

a) 10 “Fulano” “Av. Torres de Oliveira, 330”

FALSO

3.1415

b) 10 ‘Fulano’ ‘Av. Torres de Oliveira, 330’

VERDADEIRO

3

c) 10 Fulano Av. Torres de Oliveira, 330

FALSO

3.14159265358979

d) 10 Fulano Av. Torres de Oliveira, 330

VERDADEIRO

3.1415

e) 10 Fulano Av. Torres de Oliveira, 330

VERDADEIRO

3

Resposta

- Dado o algoritmo abaixo, escolha a opção que representa a saída correta.

Algoritmo "Exercicio4"

Var

num : inteiro
falsoverdadeiro : logico
nome, endereco : caractere
num_pi : real const

d) 10 Fulano Av. Torres de Oliveira, 330
VERDADEIRO
3.1415

Inicio

num_pi <- 3.1415
num <- 10
falsoverdadeiro <- ((num_pi < 3.0) ou (num >= (18 mod 5)))
nome <- " Fulano "
endereco <- "Av. Torres de Oliveira, 330"

escreval(num, nome, endereco)

escreval(falsoverdadeiro)

escreval(num_pi)

Fimalgoritmo

ATÉ A PRÓXIMA!