

Disciplina:	Introdução à Computação
Curso:	Matemática Aplicada e Computacional
Prof.:	Daniel Ventura
Grupo:	ALEKSON NICAEL DOS SANTOS ARAÚJO GUSTAVO ROSA RODRIGUES HENRIQUE HONÓRIO DA SILVA

## Lógica de Programação

**1.** Elabore um algoritmo que mova quatro discos de uma Torre de Hanói, que consiste em três hastes (a, b, c), uma das quais serve de suporte para quatro discos de tamanhos diferentes (1, 2, 3, 4), 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 menor. O objetivo é transferir os quatro discos para outra haste. por exemplo, uma instância do problema seriam os discos da haste 'a' e o objetivo seria a transferência dos discos para a torre 'c'.

**algoritmo** torre\_de\_hanoi

**inicio**

**inteiro:** disco, n;

**caractere:** origem, destino, auxiliar;

n ← 4;

origem ← "a";

destino ← "b";

auxiliar ← "c";

disco ← 1;

**se** disco ≤ n **então**

**escreva**("Mover disco ", disco, " da haste ", origem, " para a haste ", destino, "\n");

    // Troca as hastes para a próxima iteração

    disco ← disco \* 2;

**se** disco = 0 **então**

**se** origem = "a" **então**

```

        origem ← "b";
    senão
        origem ← "a";
    fimse
senão
    se destino = "c" então
        destino ← "b";
    senão
        destino ← "c";
    fimse
fimse

    disco ← disco + 1;
fim

```

**2.** Considerando uma máquina de calcular com apenas as operações de adição e subtração, defina os algoritmos para a solução das seguintes expressões:

a)  $11 \times 5$

**algoritmo** multiplicacao\_11\_por\_5

**inicio**

**inteiro:** resultado;

    resultado ← 0;

    resultado ← resultado + 11;

    resultado ← resultado + 11;

    resultado ← resultado + 11;

    resultado ← resultado + 11;

    resultado ← resultado + 11;

**escreva**("Resultado: ", resultado);

**fim**

b)  $22 \times 13$

**algoritmo** multiplicacao\_22\_por\_13

**inicio**

**inteiro:** resultado;

    resultado ← 0;

```
escreva("Resultado: ", resultado);  
fim
```

**algoritmo** divisao\_60\_por\_6

```
dividendo ← dividendo - divisor;  
resultado ← resultado + 1;  
dividendo ← dividendo - divisor;  
resultado ← resultado + 1;  
dividendo ← dividendo - divisor;  
resultado ← resultado + 1;  
dividendo ← dividendo - divisor;  
resultado ← resultado + 1;  
dividendo ← dividendo - divisor;  
resultado ← resultado + 1;
```

```
    escreva("Resultado: ", resultado);  
fim
```

d)  $176 \div 8$

**algoritmo** divisao\_176\_por\_8

**inicio**

**inteiro:** dividendo, divisor, resultado;

dividendo  $\leftarrow$  176;

divisor  $\leftarrow$  8;

resultado  $\leftarrow$  0;

// Realiza subtrações sucessivas do dividendo pelo divisor até que o dividendo seja menor que o divisor

// A cada subtração bem-sucedida, incrementa o resultado

dividendo  $\leftarrow$  dividendo - divisor;

resultado  $\leftarrow$  resultado + 1;

dividendo  $\leftarrow$  dividendo - divisor;

resultado  $\leftarrow$  resultado + 1;

dividendo  $\leftarrow$  dividendo - divisor;

resultado  $\leftarrow$  resultado + 1;

dividendo  $\leftarrow$  dividendo - divisor;

resultado  $\leftarrow$  resultado + 1;

dividendo  $\leftarrow$  dividendo - divisor;

resultado  $\leftarrow$  resultado + 1;

dividendo  $\leftarrow$  dividendo - divisor;

resultado  $\leftarrow$  resultado + 1;

dividendo  $\leftarrow$  dividendo - divisor;

resultado  $\leftarrow$  resultado + 1;

dividendo  $\leftarrow$  dividendo - divisor;

resultado  $\leftarrow$  resultado + 1;

```
    escreva("Resultado: ", resultado);  
fim
```

e)  $2^{10}$

**algoritmo** potencia\_de\_2\_elevado\_10

**inicio**

**inteiro:** resultado, base, expoente, contador;

```

base ← 2;
expoente ← 10;
resultado ← base;

// Realiza multiplicações sucessivas para obter a potência
// Utilizando a propriedade de que  $2^{10} = 2 * 2 * 2 * 2 * 2 * 2 * 2 * 2 * 2 * 2$ 
resultado ← resultado + resultado + resultado + resultado + resultado + resultado +
resultado + resultado + resultado + resultado;

escreva("Resultado: ", resultado)
fim

```

## Fundamentos para Algoritmos

**3.** Supondo que a, b, c são variáveis do tipo inteiro com valores iguais a 5, 10, -8, respectivamente, e d é uma variável do tipo real com valor igual a 1,5, quais os resultados das expressões aritméticas a seguir:

a)  $2 * a \bmod 3 - c$

$$2 * 5 \bmod 3 - (-8)$$

$$10 \bmod 3 + 8$$

$$1 + 8 = 9$$

b)  $rad(-2 * c) \div 4$

$$rad(-2 * (-8)) \div 4$$

$$rad(16) \div 4$$

$$4 \div 4 = 1$$

c)  $((20 \div 3) \div 3) + pot(2, 8) / 2$

$$((20 \div 3) \div 3) + pot(2, 8) / 2$$

$$(6 \div 3) + pot(2, 8) / 2$$

$$2 + 256 / 2$$

$$2 + 128 = 130$$

d)  $(30 \bmod 4 * pot(3, 3)) * -1$

$$(2 * pot(3, 3)) * -1$$

$$(2 * 27) * -1 = -54$$

e)  $pot(-c, 2) + (d * 10) / a$

$$pot(-(-8), 2) + (1,5 * 10) / 5$$

$$64 + 15 / 5$$

$$64 + 3 = 67$$

$$\begin{aligned}
 & f) \text{rad}(\text{pot}(a, b/a)) + c * d \\
 & \text{rad}(\text{pot}(5, 10/5)) + (-8) * 1,5 \\
 & \text{rad}(\text{pot}(5, 2)) - 12 \\
 & \text{rad}(25) - 12 \\
 & 5 - 12 = -7
 \end{aligned}$$

**4.** Determine os resultados obtidos na avaliação das expressões lógicas seguintes.

Considere que os valores iniciais das variáveis são:

a.  $x = 1, y = 2, z = 5$

c.  $x = 2, z = 9, \text{Nome} = \text{"MARIA"}, \text{Sim} = \text{False}$

b.  $x = 4, y = 3, z = 1$

d.  $x = 4, y = 9, \text{Sim} = \text{True}, \text{Nome} = \text{"ANA"}$

a)  $x * x + y > z$

$$1 * 1 + 2 > 5$$

$$1 + 2 > 5$$

$$3 > 5 \implies \text{False}$$

b)  $x * x + y > z$

$$4 * 4 + 3 > 1$$

$$16 + 3 > 1$$

$$19 > 1 \implies \text{True}$$

c)  $(\text{Nome} = \text{"JORGE"}) \text{ e } \text{Sim} \text{ ou } (\text{rad}(x) < z * 10)$

$$(\text{False}) \text{ e } \text{False} \text{ ou } (\text{rad}(2) < 9 * 10)$$

$$\text{False} \text{ ou } (\text{rad}(2) < 90)$$

$$\text{False} \text{ ou } \text{True} \implies \text{True}$$

d)  $(\text{rad}(x) > \text{rad}(y)) \text{ ou } \text{não Sim} \text{ e } (\text{Nome} = \text{"PEDRO"})$

$$(\text{rad}(4) > \text{rad}(9)) \text{ ou } \text{False} \text{ e } (\text{False})$$

$$(\text{rad}(4) > \text{rad}(9)) \text{ ou } \text{False} \text{ e } (\text{False})$$

$$\text{False} \text{ ou } \text{False} \implies \text{False}$$

**5.** Escreva as expressões algébricas em forma de programa.

Exemplo: Expressão:  $x + vy$

Algoritmo:  $x + v * y$

a)  $a + bc + d$

$$a + b * c + d$$

b)  $32 \sqrt{2b-c}$

$$32 * \text{rad}(2) * b - c$$

$$\text{c) } (a+b) c + d (a-2b) \\ (a + b) * c + d * (a - 2 * b)$$

$$\text{d) } (a+b) - \sqrt{2}(b-c) \\ (a + b) - \text{rad}(2) * (b - c)$$

$$\text{e) } [2a + (c-d)^2]^2 \\ [2 * a + (c - d) * 2]^2$$

$$\text{f) } (x+y)(x-y) \\ (x + y) * (x - y)$$

**6.** Considere as variáveis declaradas abaixo:

inteiro: d, y, p, q, r;

real: a, b, c, s, z;

Indique qual o resultado das expressões aritméticas, onde  $a = 3.0$ ;  $b = 2.0$ ;  $c = 0.5$ ;  $s = 9.0$ ;  $z = 12.0$ ;

$d = 16$ ;  $y = 2$ ;  $p = 4$ ;  $q = 6$ ;  $r = 24$ ;

$$\text{a) } x + y - z * a \\ x + 2 - 12.0 * 3.0 \\ x + 2 - 36.0 \\ x = 34.0$$

$$\text{b) } d \text{ div } y \\ 16 \text{ div } 2 = 8$$

$$\text{c) } y \text{ mod } d \\ 2 \text{ mod } 16 = 2$$

$$\text{d) } p * (r \text{ mod } q) - q / 2 \\ 4 * (24 \text{ mod } 6) - 6 / 2 \\ - 6 / 2 = -3$$

$$\text{e) } \text{rad}(\text{pot}(a - b * y - d, 2)) \\ \text{rad}(\text{pot}(3.0 - 2.0 * 2 - 16, 2)) \\ \text{rad}(\text{pot}(-17.0, 2)) \\ \text{rad}(289.0) = 17.0$$

$$\text{f) } ((z/a) + b * a) - d$$

$$((12.0/3.0) + 2.0 * 3.0) - 16$$

$$(4.0 + 6.0) - 16$$

$$10.0 - 16 = -6.0$$

**7.** Determine os resultados obtidos na avaliação das expressões lógicas seguintes, sabendo que a, b, c, contêm, respectivamente, 2, 7, 3.5, e que existe uma variável lógica l cujo valor é falso (False)

a)  $b = a * c$  e ( l ou True)

$7 = 2 * 3.5$  e (False ou True)

True e True  $\implies$  True

b)  $b > a$  ou  $b = \text{pot}(a,a)$

$7 > 2$  ou  $7 = \text{pot}(2,2)$

True ou False  $\implies$  True

c) l e  $b \text{ div } a \geq c$  ou não  $a \leq c$

False e  $7 \text{ div } 2 \geq 3.5$  ou não True

False e True ou False

False e True  $\implies$  False

d) não l ou True e  $\text{rad}(a + b) \geq c$

não False ou True e  $\text{rad}(2 + 7) \geq 3.5$

True ou True e  $3 \geq 3.5$

True e False  $\implies$  False

e)  $b / a = c$  ou  $b / a \neq c$

$7 / 2 = 3.5$  ou  $7 / 2 \neq 3.5$

True ou False  $\implies$  True

f) l ou  $\text{pot}(b,a) \leq c * 10 + a * b$

False ou  $\text{pot}(7,2) \leq 3.5 * 10 + 2 * 7$

False ou  $49 \leq 49$

False ou True  $\implies$  True

## Comandos Básicos

**8.** Faça um programa que receba as 4 notas de um aluno durante o ano, calcule e imprima sua média final (média aritmética).

**algoritmo** calcular\_media\_aluno



**inicio**

**real:** nota1, nota2, nota3, nota4, media;

// Solicita ao usuário que insira as quatro notas

**escreva**("Digite a primeira nota: ");

**leia**(nota1);

**escreva**("Digite a segunda nota: ");

**leia**(nota2);

**escreva**("Digite a terceira nota: ");

**leia**(nota3);

**escreva**("Digite a quarta nota: ");

**leia**(nota4);

// Calcula a média das notas

media ← (nota1 + nota2 + nota3 + nota4) / 4;

// Imprime a média final

**escreva**("A média final do aluno é: ", media);

**fim**

**9.** Faça um programa que receba o valor do salário de um funcionário e o valor do salário mínimo.

Calcule e imprima quantos salários mínimos ganha esse funcionário.

**algoritmo** calcular\_salarios\_minimos

**inicio**

**real:** salario\_funcionario, salario\_minimo, qtd\_salarios\_minimos;

// Solicita ao usuário que insira o valor do salário do funcionário

**escreva**("Digite o valor do salário do funcionário: ");

**leia**(salario\_funcionario);

// Solicita ao usuário que insira o valor do salário mínimo

**escreva**("Digite o valor do salário mínimo: ");

**leia**(salario\_minimo);

// Calcula quantos salários mínimos o funcionário ganha

qtd\_salarios\_minimos ← salario\_funcionario / salario\_minimo;

// Imprime a quantidade de salários mínimos que o funcionário ganha

```
    escreva("O funcionário ganha ", qtd_salarios_minimos, " salários mínimos.");  
fim
```

**10.** Faça um programa que receba a medida de ângulo em graus, calcule e imprima o seno e o cosseno.

**algoritmo** calcular\_seno\_cosseno

**inicio**

```
    real: angulo_graus, angulo_radianos, seno, cosseno, termo_seno, termo_cosseno;
```

```
    inteiro: fatorial_2, fatorial_3, fatorial_4, fatorial_5;
```

```
    // Solicita ao usuário que insira a medida do ângulo em graus
```

```
    escreva("Digite a medida do ângulo em graus: ");
```

```
    leia(angulo_graus);
```

```
    // Converte o ângulo de graus para radianos
```

```
    angulo_radianos ← angulo_graus * PI / 180;
```

```
    // Inicializa os valores de seno e cosseno
```

```
    seno ← 0;
```

```
    cosseno ← 0;
```

```
    // Calcula os fatoriais necessários
```

```
    fatorial_2 ← 1 * 2;
```

```
    fatorial_3 ← fatorial_2 * 3;
```

```
    fatorial_4 ← fatorial_3 * 4;
```

```
    fatorial_5 ← fatorial_4 * 5;
```

```
    // Calcula os termos para o seno e o cosseno
```

```
    termo_seno ← angulo_radianos;
```

```
    termo_cosseno ← 1;
```

```
    seno ← termo_seno;
```

```
    cosseno ← termo_cosseno;
```

```
    termo_seno ← - (angulo_radianos^3) / fatorial_3;
```

```
    termo_cosseno ← (angulo_radianos^2) / fatorial_2;
```

```
    seno ← seno + termo_seno;
```

```
    cosseno ← cosseno + termo_cosseno;
```

```
    termo_seno ← (angulo_radianos^5) / fatorial_5;
```

```

termo_cosseno ← - (angulo_radianos^4) / fatorial_4;

seno ← seno + termo_seno;
cosseno ← cosseno + termo_cosseno;

// Imprime o seno e o cosseno do ângulo
escreva("O seno do ângulo é: ", seno, " e o cosseno do ângulo é: ", cosseno);
fim

```

**11.** Dado as funções abaixo, teste o programa a seguir e verifique os resultados

<b>copia</b> (cadeia, posição, número)	Copia da cadeia, a partir da posição dada, o número de caracteres estipulados.
<b>comp</b> (cadeia)	Mostra o número de caracteres da cadeia.
<b>pos</b> (cadeia1, cadeia2)	Mostra em que posição a cadeia1 aparece dentro da cadeia2.
<b>remove</b> (cadeia, posição, número)	Apaga da cadeia, a partir da posição dada, o número de caracteres estipulados.
<b>insere</b> (cadeia1, cadeia2, posição)	Insere na cadeia2 a cadeia1 a partir da posição dada.
<b>concat</b> (cadeia1, cadeia2, cadeia3)	concatena as cadeias

```

constante Lista = "PROCESSAMENTODEDADOS";
           Seq = "PPPP";
           Seq1 = "DE";

```

**inicio**

```

inteiro: k, m;
caracter: a, x, y, z, w;

k ← pos(Seq1, Lista);
m ← comp(Seq);
x ← copia(Lista, 7, m+2);
y ← copia(Lista, k-10, m-1);
z ← concat(Seq, Lista);
a ← concat(Seq, Seq1, Seq);
w ← concat("***", copia(Lista,1,2), "***");
escreva(x);
escreva(y);
escreva(z);
escreva(a);
escreva(w);

```

**fim.**

A execução do código acima, tem o seguinte resultado:

OCESSAMENTODEDADOS  
OCESSAMENTODEDA  
PPPPOCESSAMENTODEDADOS  
PPPPOCESSAMENTODEDADOSDEPPP  
\*\*\*PR\*\*\*

Resultados das operações de manipulação de strings conforme definidas no código

## 12. Dadas as declarações:

**inteiro:** Num;  
**real:** Soma, x;  
**caracter:** Simbolo, Nome, Cor, Dia;  
**logico:** Cod, Teste, Tudo;

Assinale os comandos de atribuição inválidos:

(X) Simbolo ← 5;	( ) Soma ← Num + 2 * x;
(X) Teste ← Cor;	(X) Tudo ← Soma;
( ) Cor ← "preto";	( ) x ← x + 1;
(X) Num ← "*ABC*";	( ) Dia ← "SEGUNDA";

**13.** Faça um programa que receba o nome e o salário de um funcionário, calcule e imprima o nome e o valor do imposto de renda a ser pago, sabendo que o imposto equivale a 5% do salário.

**algoritmo** calcular\_imposto\_renda

**inicio**

**caractere:** nome;  
**real:** salario, imposto;

// Solicita ao usuário que insira o nome do funcionário

**escreva**("Digite o nome do funcionário: ");

**leia**(nome);

// Solicita ao usuário que insira o salário do funcionário

**escreva**("Digite o salário do funcionário: ");

**leia**(salario);

```

// Calcula o valor do imposto de renda (5% do salário)
imposto ← salario * 0.05;

// Imprime o nome do funcionário e o valor do imposto de renda a ser pago
escreva("Nome do funcionário: ", nome, "\n");
escreva("Imposto de renda a ser pago: ", imposto);
fim

```

**14.** Escreva um programa que calcule e imprima a área de um retângulo. O programa deve solicitar ao usuário as informações necessárias para esse cálculo.

**algoritmo** calcular\_area\_retangulo

**inicio**

**real:** comprimento, largura, area;

```

// Solicita ao usuário que insira o comprimento do retângulo
escreva("Digite o comprimento do retângulo: ");
leia(comprimento);

```

```

// Solicita ao usuário que insira a largura do retângulo
escreva("Digite a largura do retângulo: ");
leia(largura);

```

```

// Calcula a área do retângulo
area ← comprimento * largura;

```

```

// Imprime a área do retângulo
escreva("A área do retângulo é: ", area);

```

**fim**

**15.** Escreva um programa que imprima na tela a mensagem “Hello World!”.

**algoritmo** hello\_world

**inicio**

```

// Imprime a mensagem "Hello World!" na tela
escreva("Hello World!");

```

**fim**

**16.** Escreva um programa que leia 2 valores A e B e calcule o valor de C, sabendo que  $C = (A + B)$

\* B. O programa deve imprimir os valores de A, B e C.

**algoritmo** calcular\_c

**inicio**

**real:** A, B, C;

// Solicita ao usuário que insira o valor de A

**escreva**("Digite o valor de A: ");

**leia**(A);

// Solicita ao usuário que insira o valor de B

**escreva**("Digite o valor de B: ");

**leia**(B);

// Calcula o valor de C

$C \leftarrow (A + B) * B;$

// Imprime os valores de A, B e C

**escreva**("Valor de A: ", A, "\n");

**escreva**("Valor de B: ", B, "\n");

**escreva**("Valor de C: ", C);

**fim**