

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) 11×5

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) 22×13

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;

 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 ← 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

```

e) 2^{10}

algoritmo potencia_de_2_elevado_10

inicio

inteiro: resultado, base;

```

base ← 2;
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) $rad(-2 * c) \text{ div } 4$
 $rad(-2 * (-8)) \text{ div } 4$
 $rad(16) \text{ div } 4$
 $4 \text{ div } 4 = 1$
- c) $((20 \text{ div } 3) \text{ div } 3) + pot(2, 8) / 2$
 $((20 \text{ div } 3) \text{ div } 3) + pot(2, 8) / 2$
 $(6 \text{ div } 3) + pot(2, 8) / 2$
 $2 + 256 / 2$
 $2 + 128 = 130$
- d) $(30 \text{ mod } 4 * pot(3, 3)) * -1$
 $(2 * pot(3, 3)) * -1$
 $(2 * 27) * -1 = -54$
- e) $pot(-c, 2) + (d * 10) / a$
 $pot(-(-8), 2) + (1,5 * 10) / 5$
 $64 + 15 / 5$
 $64 + 3 = 67$
- f) $rad(pot(a, b/a)) + c * d$
 $rad(pot(5, 10/5)) + (-8) * 1,5$
 $rad(pot(5, 2)) - 12$
 $rad(25) - 12$
 $5 - 12 = -7$

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

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

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

False ou True \implies True

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

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

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

False ou False \implies 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) $32 \sqrt{2b-c}$

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

c) $(a+b) c + d (a-2b)$

$(a + b) * c + d * (a - 2 * b)$

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

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

e) $[2a + (c-d)^2]^2$

$[2 * a + (c - d) * 2]^2$

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 $a = 3.0$; $b = 2.0$; $c = 0.5$; $s = 9.0$; $z=12.0$;

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

a) $x + y - z * a$

$x + 2 - 12.0 * 3.0$

$x + 2 - 36.0$

$x - 34.0$

$$x = 34.0$$

b) $d \text{ div } y$

$$16 \text{ div } 2 = 8$$

c) $y \text{ mod } d$

$$2 \text{ mod } 16 = 2$$

d) $p * (r \text{ mod } q) - q / 2$

$$4 * (24 \text{ mod } 6) - 6 / 2$$

$$- 6 / 2 = -3$$

e) $\text{rad}(\text{pot}(a - b * y - d, 2))$

$$\text{rad}(\text{pot}(3.0 - 2.0 * 2 - 16, 2))$$

$$\text{rad}(\text{pot}(-17.0, 2))$$

$$\text{rad}(289.0) = 17.0$$

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) $b = a * c \text{ e } (l \text{ ou True})$

$$7 = 2 * 3.5 \text{ e } (\text{False ou True})$$

$$\text{True e True} \implies \text{True}$$

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

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

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

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

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

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

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

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

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

$$\text{True ou True e } 3 \geq 3.5$$

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

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 \leftarrow (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

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

1. Define a constante Lista como "PROCESSAMENTODEDADOS".
2. Define as variáveis Seq como "PPPP" e Seq1 como "DE".
3. Encontra a posição da sequência Seq1 dentro da string Lista e armazena em k.
4. Calcula o comprimento da sequência Seq e armazena em m.
5. Extrai uma substring de Lista começando na posição 7 e com comprimento m + 2 e armazena em x.
6. Extrai uma substring de Lista começando na posição k - 10 e com comprimento m - 1 e armazena em y.
7. Concatena as sequências Seq e Lista e armazena em z.
8. Concatena as sequências Seq, Seq1 e Seq e armazena em a.
9. Concatena " ***", uma substring de Lista começando na posição 1 e com comprimento 2, e " ***" e armazena em w.
10. Imprime as variáveis x, y, z, a e w.

A execução do código acima, tem o seguinte resultado:

```
OCESSAMENTODEDADOS
OCESSAMENTODEDA
PPPOCESSAMENTODEDADOS
PPPOCESSAMENTODEDADOSDEPPP
***PR***
```

Resultados das operações de manipulação de strings conforme definidas no código

12. Dadas as declarações:

inteiro: Num;

real: Soma, x;

character: Simbolo, Nome, Cor, Dia;

logico: Cod, Teste, Tudo;

Assinale os comandos de atribuição inválidos:

- | | |
|-------------------------------|------------------------------------|
| (X) Simbolo \leftarrow 5; | () Soma \leftarrow Num + 2 * x; |
| (X) Teste \leftarrow Cor; | (X) Tudo \leftarrow Soma; |
| () Cor \leftarrow "preto"; | () x \leftarrow x + 1; |
| (X) Num \leftarrow "*ABC*"; | () Dia \leftarrow "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 \leftarrow 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
```