Alxgoritmo

Manassés Ribeiro <https://meet.google.com/xun-rfci-odo>

[manasses.ribeiro@ifc.edu.br](mailto:manasses.ribeiro@ifc.edu.br) – Pessoal [cienciadacomputacao.grad.videira@ifc.edu.br](mailto:cienciadacomputacao.grad.videira@ifc.edu.br) – Coordenação do curso



**Aula 1 - Introdução a lógica de programação**

**Noções de lógica**

* A lógica estuda e ensina a colocar ordem no pensamento.

Ex:

todo mamífero é um animal.

Todo cavalo é um mamífero,

Logo, todo cavalo é um animal.

**Formas de representar um algoritmo**

**TROCA DE LÂMPADA**

* Pegar uma escada;
* Posicionar a escada embaixo da lâmpada;
* Buscar uma lâmpada nova;
* Subir na escada;
* Retirar a lâmpada velha;
* Colocar a lâmpada nova.

**TROCA DE LÂMPADA COM TESTE**

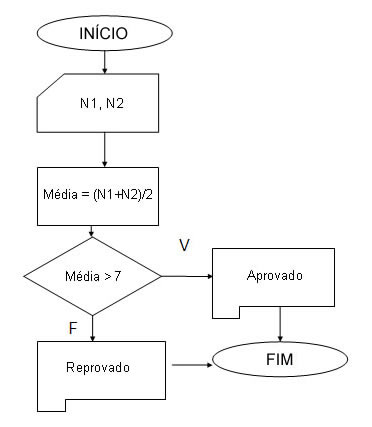
* Pegar uma escada;
* Posicionar a escada embaixo da lâmpada;
* Buscar uma lâmpada nova;
* Acionar o interruptor;
* Se a lâmpada não acender, então
  + Subir na escada;
  + Retirar a lâmpada queimada;
  + Colocar a lâmpada nova.

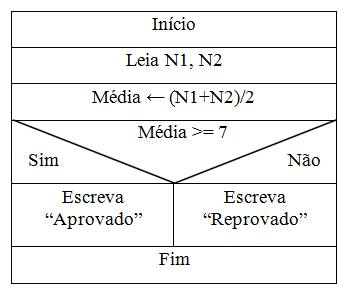
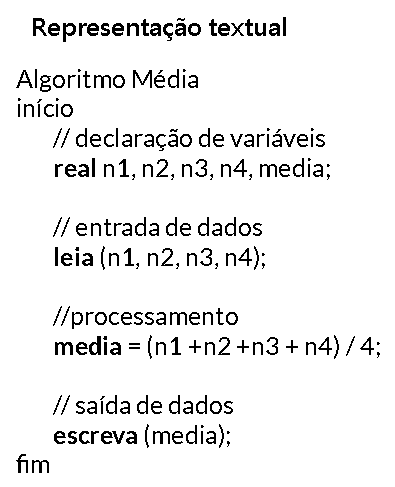
**TROCA DE LÂMPADA COM TESTE NO INÍCIO**

* Acionar o interruptor;
* Se a lâmpada não acender, então
  + Pegar uma escada
  + Posicionar a escada embaixo da lâmpada;
  + Buscar uma lâmpada nova;
  + Subir na escada;
  + Retirar a lâmpada queimada;
  + Colocar a lâmpada nova.

**TROCA DE LÂMPADA COM TESTE E CONDIÇÃO DE PARADA**

* Acionar o interruptor;
* Se a lâmpada não acender, então
  + Pegar uma escada;
  + Posicionar a escada embaixo da lâmpada;
  + Buscar uma lâmpada nova;
  + Subir na escada;
  + Retirar a lâmpada queimada;
  + Colocar a lâmpada nova.
  + Enquanto a lâmpada não acender, faça
    - Retirar a lâmpada queimada;
    - Colocar uma lâmpada nova.





**Exercícios**

1. Três senhoras, dona Branca, dona Rosa e dona Violeta passeavam pelo parque quando dona Rosa disse: “Não é curioso que estejamos usando vestidos de cores branca, rosa e violeta, embora nenhuma de nós esteja usando um vestido de cor igual ao seu próprio nome. "Uma simples coincidência" respondeu a senhora com o vestido violeta. Diga. Qual a cor do vestido de cada senhora?

|  |  |  |
| --- | --- | --- |
| Nome | Cor do vestido | Por quê? |
| Rosa | Branca | Não pode usar rosa pois é o próprio nome, e não usa violeta pois é a cor do vestido de quem a respondeu |
| Branca | Violeta | Só sobra a cor violeta |
| Violeta | Branca | Só sobra a cor branca |

1. Um homem precisa atravessar um rio com um barco que possui capacidade de transportar apenas ele mesmo e mais uma de suas três cargas, que são: um lobo, um bode e um maço de alfafa. O que o homem deve fazer para conseguir atravessar o rio sem perder suas cargas?

**Inicio**

Atravessa com o bode 🡪

Volta o homem 🡨

Atravessa o homem e o lobo 🡪

Volta o homem e o bode 🡨

Atravessa o homem e a alfafa 🡪

Volta o homem 🡨

Atravessa o homem e o bode 🡪

**Fim**

1. Elabore um algoritmo que mova três discos de uma Torre de Hanói, que consiste em três hastes (a-b-c), uma das quais serve de suporte para três discos de tamanhos diferentes (1-2-3), 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 três discos para outra haste.

**Início**

Mover o disco 1 para a haste b

Mover o disco 2 para a haste c

Mover o disco 1 para a haste c

Mover o disco 3 para a haste b

Mover o disco 1 para a haste a

Mover o disco 2 para a haste b

Mover o disco 1 para a haste b

**Fim**

**Aula 2 - Tipos Primitivos de Dados, Variáveis e Operadores**

**Tipos primitivos de dados**

* Inteiro:

Informação numérica pertencente aos inteiros. (negativo, nulo e positivo).

Ex: -2, 0, 10.

* Real:

Informação numérica que pertença aos reais. (negativo, nulo e positivo).

Ex: 1.75, 0,35, 82.50.

* Caractere:

Informação composta de caracteres alfanuméricos.

Ex: numéricos (0...9), alfabético (A...Z, a...z) e especiais (#, ?, !, @)

* Lógico: Informação que pode assumir apenas duas situações (Verdadeiro e Falso)

Ex: A luz está acesa? (Sim) ou (Não)

**Constante e variável**

* Constante:

Um dado que não sofre alteração com o tempo.

Ex: PI (π)

* Variável:

Um dado que pode ter seu valor alterado no passar do tempo.

Ex: Salário, valor do dólar.

**Formatação dos identificadores**

* Sempre começam com um caractere alfabético (preferencialmente minúsculo)
* Podem ser seguidos por cais caracteres alfabéticos ou numéricos
* Não devem ser usados caracteres especiais
* Em constantes, usa-se tudo em MAIUSCULO

**Declaração de variáveis**

* **Caractere**

Ex: Nome;

* **Real**

Ex: Salário;

* **Inteiro**

Ex: Idade;

* **Lógico**

Ex: Resposta;

**Expressões aritméticas**

|  |  |  |
| --- | --- | --- |
| Operador | Função | Exemplos |
| + | Adição | 2 + 3, x + y |
| - | Subtração | 4 – 2, n – m |
| \* | Multiplicação | 3 \* 4, a \* b |
| / | Divisão | 10 / 2, x1 / x2 |
| Pot(x,y) | Potenciação | pot(2,3) |
| Rad(x) | Radiciação | rad(9) |
| mod | Resto da divisão | 9 mod 4 resulta em 1  27 mod 5 resulta em 2 |
| div | Quociente da divisão | 9 div 4 resulta em 2  27 div 5 resulta em 5 |

**Prioridades das expressões**

|  |  |
| --- | --- |
| Prioridade | Operadores |
| 1ª | Parênteses mais internos |
| 2ª | pot, rad |
| 3ª | \*, /, div, mod |
| 4ª | +, - |

**Exemplos**

1. 5 + 9 + 7 + 8 / 4

5 + 9 + 7 + 2

23

1. 1 – 4 \* 3 / 6 – pot(3,2)

1 – 4 \* 3 / 6 – 9

1 – 12 / 6 – 9

1 – 2 – 9

-10

1. pot(5,2) – 4 / 2 + rad(1 + 3 \* 5) / 2 11 linhas

pot(5,2) – 4 / 2 + rad(1 + 15) / 2

pot(5,2) – 4 / 2 + rad(16) / 2

25 – 4 / 2 + 4 / 2

25 – 2 + 2

25

**Exercícios de fixação**

Supondo que A, B e C são variáveis de tipo inteiro, com valores iguais a 5, 10 e -8, respectivamente, e uma variável real D, com valor de 1.5, quais os resultados das expressões aritméticas a seguir?

1. **2 \* A mod 3 – C**

2 \* 2 – ( – 8)

2 \* 2 + 8

4 + 8

12

1. **rad(–2 \* C) div 4**

rad(16) div 4

4 div 4

1

1. **( (20 div 3) div 3) + pot(8, 2) / 2**

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

2 + pot( 8, 2) / 2

2 + 64 / 2

2 + 32

34

1. **(30 mod 4 \* pot(3 , 3) ) \*– 1**

(30 mod 4 \* 27) \* -1

(2 \* 27) \* -1

(54) \* -1

-54

1. **pot( – C, 2) + (D \* 10) / A**

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

pot(8,2) + (15)/5

64 + 3

67

1. **rad( pot( A, B / A) ) + C \* D**

rad( pot( 5, 10 / 5) ) + ( - 8) \* 1.5

rad( pot( 5, 2) ) + (-8) \* 1.5

rad( 25 ) + (-8) \* 1.5

5 + (-8) \* 1.5

5 – 12

- 7

**Operadores relacionais**

|  |  |  |
| --- | --- | --- |
| **Operador** | **Função** | **Exemplos** |
| **==** | Igual a | 3 == 3, x == y |
| **>** | Maior que | 5 > 3, x > y |
| **<** | Menor que | 3 < 5, x < y |
| **>=** | Maior ou igual a | 5 >= 3, x >= y |
| **<=** | Menor ou igual a | 3 <= 5, y <=x |
| **!=** | Diferente de | 3 != 5, x != y |

**O resultado SEMPRE será um valor lógico. (True, False)**

**Exercícios**

1. 2 \* 4 == 24 / 3 🡪 8 == 8 🡪 V
2. 15 mod 4 < 19 mod 6 🡪 3 < 1 🡪 F
3. 3 \* 5 div 4 <= pot( 3, 2 ) / 0.5 🡪 3 <= 4.5 🡪 V
4. 2 + 8 mod 7 >= 3 \* 6 – 15 🡪 3 >= 3 🡪 V

**Operadores lógicos**

* Não

Negação

* E

Conjunção

* Ou

Disjunção

**Prioridade entre operadores lógicos**

Não --> E --> Ou

**Exercícios**

1. 2 < 5 e 15/3 == 5

V e 5 == 5

V e V

V

1. 2 < 5 ou 15/3 == 5

V ou 5 == 5

V ou V

V

1. F ou 20 div (18/3) != (21/3) div 2

F ou 20 div 6 != 7 div 2

F ou 3 != 3

F ou F

F

1. não V ou pot (3,2) / 3 < 15 - 35 mod 7

F ou 9 / 3 < 15 – 0

F ou 3 < 15

F ou V

V

**Prioridade entre todos os operadores**

|  |  |
| --- | --- |
|  | Prioridade |
| 1º | Parênteses mais internos |
| 2º | Operadores aritméticos (mod, rad, div, ...) |
| 3º | Operadores relacionais (==, <=, <, ...) |
| 4º | Operadores lógicos (Não, E, Ou) |

**Exercícios**

Determine o valor das expressões lógicas abaixo, considerando que A, B e C contêm, respectivamente, 2, 7, 3.5 e que para a variável lógica L seu valor é falsidade (F)

1. **B == A \* C e (L ou V)**

7 == 2 \* 3.5 e (F ou V)

7 == 7 e V

V e V

V

1. **B > A ou B == pot(A, A)**

7 > 2 ou 7 == pot(2,2)

7 > 2 ou 7 == 4

V ou F

V

1. **L e B div A >= C ou não A <= C**

F e 7 div 2 >= 3.5 ou não 2 <= 3.5

F e 7 div 2 >= 3.5 ou não V

F e 7 div 2 >= 3.5 ou F

F e 3 >= 3.5 ou F

F e F ou F

F ou F

F

1. **não L ou V e rad(A + B) >= C**

não F ou V e 3>=3.5

V ou V e F

V ou F

V

1. **B/A == C ou B/A != C**

7 / 2 == 3.5 ou 7 / 2 != 3.5

3.5 == 3.5 ou 3.5 != 3.5

V ou F

V

1. **L ou pot (B, A) <= C \* 10 + A \* B**

F ou 49 <= 3.5 \* 10 + 2 \* 7

F ou 49 <= 35 + 14

F ou 49 <= 49

F ou V

V

**Aula 3 - Comandos básicos e estrutura do algoritmo**

**Comandos de atribuição**

Ex:

Inteiro **idade**, x;

Lógico trueOrfalse;

Idade = 25;

trueOrfalse = True;

**Exercício**:

Encontre os erros dos seguintes comandos de atribuição:

lógico a;

real b, c;

inteiro d;

a = b == c;

d = b;

c + 1 = b + c;

c e b = 3.5;

b = pot(6,2) / 3 <= rad(9) \* 4;

**Entrada, saída e bloco**

* **Entrada**

É utilizado para preencher o valor de uma variável proveniente de um meio externo.

Ex: leia (x); leia (a, xpto, nota);

* **Saída**

É utilizado para mostrar o resultado da operação feita

Ex: escreva (x); escreva (“Você pesa”, peso, “quilos”); escreva (a - pot(6,2) / 4);

* Bloco
  + Início

Utiliza-se no início de um bloco de operações a serem realizadas.

* + Fim

Utiliza-se para delimitar o fim de um bloco de operações a serem realizadas.

Ex:

Início //Inicio do bloco

Comando 1;

Comando 2;

...

Comando 3;

Fim //fim do bloco

**Exercício**

Considerando o algoritmo ao lado, explique o que está acontecendo em cada linha e qual é o resultado de cada ação executada.

início

inteiro x, y;

real z;

leia (x);

escreva (x, “elevado ao cubo =”, pot (x,3));

leia (y);

escreva (x + y);

z = x / y;

escreva (z);

z = z + 1;

x = (y + x ) mod 2;

escreva (x);

fim

O programa pede dois valores e retorna o cubo do primeiro, a soma dos dois valores, a divisão do 1º pelo 2º e o resto da divisão da soma deles por 2.

**Exercícios Avaliados**

1. Construa um algoritmo para calcular as raízes de uma equação do 2º grau (Ax2 + Bx + C), sendo que os valores A, B e C são fornecidos pelo usuário (considere que a equação possui duas raízes reais)

Inicio

Inteiro a, b, c, delta;

Real x1, x2;

Escreva("Informe o valor de A");

Leia(a);

Escreva("Informe o valor de B");

Leia(b);

Escreva("Informe o valor de C");

Leia(c);

delta = pot(b,2) - 4 \* a \* c;

x1 = (- b + rad(delta))/2;

x2 = (- b - rad(delta))/2;

Escreva("As raizes da sua equação são: ",x1," e ",x2,".");

Fim

1. Construa um algoritmo que, tendo como dados de entrada dois pontos quaisquer do plano cartesiano, P(x1, y1) e Q(x2, y2), escreva a distância entre eles.

Inicio

Inteiro x1, y1, x2, y2;

Real distancia;

Escreva("Informe a distância do 1º ponto no eixo X");

Leia(x1);

Escreva("Informe a distância do 1º ponto no eixo Y");

Leia(y1);

Escreva("Informe a distância do 2º ponto no eixo X");

Leia(x2);

Escreva("Informe a distância do 2º ponto no eixo Y");

Leia(y2);

distancia = rad( pot( ( x2 – x1 ), 2 ) + pot( ( y2 – y1 ), 2 ) );

Escreva("A distância entre os dois pontos é de: ",distancia);

Fim

1. Faça um algoritmo para calcular o volume de uma esfera de raio R, em que R é um valor fornecido pelo usuário. O volume de uma esfera é dado por V = 4/3 𝝿R3.

Inicio

Inteiro raio;

Real diametro;

Constante PI;

Escreva("Informe o raio da esfera que quer calcular");

Leia(raio);

diâmetro = 4 / 3 \* PI \* pot(raio,3);

Escreva(“O diâmetro da esfera é de: “, diamentro);

Fim

**Aula 4 - Estrutura de Seleção**

* Seleção simples

Exemplo:

Se (condição) então

inicio

...

...

...

fim

Fimse

Média:

Inicio

Real n1, n2, n3, n4, media;

Leia(n1, n2, n3, n4);

Media = (n1 + n2 + n3 + n4) / 4;

Escreva ( media );

Se ( media >= 7) então

Inicio

Escreva(“Aluno aprovado”);

Fim

Fimse

Fim

* Seleção composta

Exemplo:

se () então

inicio //inicio bloco verdade

Comando 1;

Comando 2;

fim

senão

início //início bloco falsidade

Comando 1;

Comando 2;

fim

fimse

Algoritmo Média

início

real n1, n2, n3, n4, media;

leia (n1, n2, n3, n4);

media = (n1 +n2 +n3 + n4) / 4;

escreva (media);

se (media >= 7.0) então

início //início bloco verdade

escreva (“Aluno aprovado!”);

fim

senão

início //início bloco falsidade

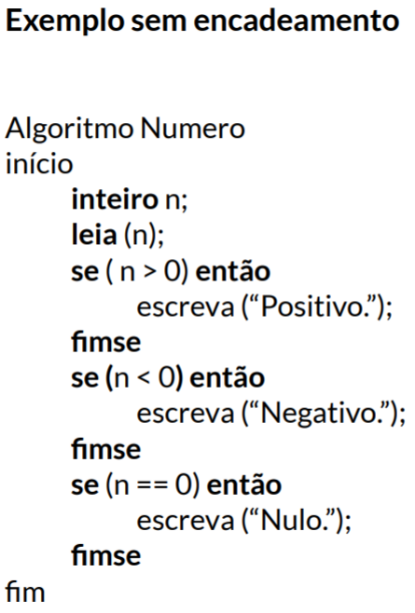
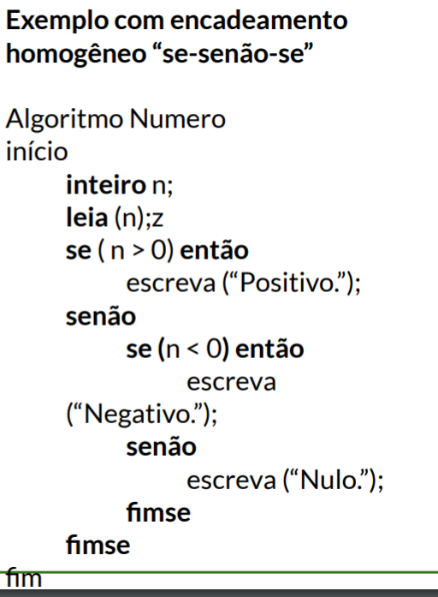
escreva (“Aluno reprovado!”);

fim

fimse

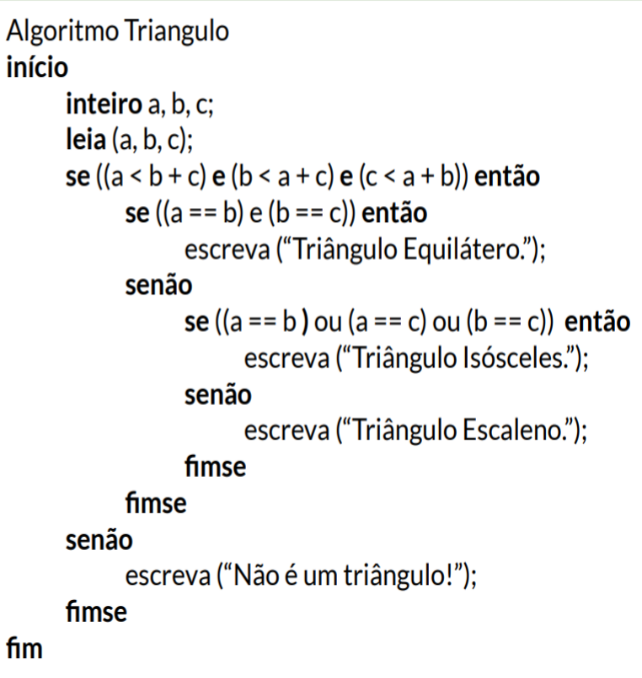
fim

* **Seleção encadeada**



* + **Seleção encadeada homogênea**

Quando a construção de uma estrutura de seleção encadeada segue um padrão lógico.

* + **Seleção encadeada heterogênea**

Quando a construção de uma estrutura de seleção encadeada não segue um padrão lógico.

Exemplo: Construa um algoritmo que leia três valores (a, b e c), verifique se os valores informados correspondem aos lados de um triângulo, e caso seja triângulo, verifique se é equilátero (três lados iguais), isósceles (dois lados iguais) ou escaleno (todos os lados diferentes).

**Exercícios**

1. Escreva um algoritmo que leia três valores inteiros e diferentes e mostre-os em ordem crescente. Utilize para tal uma seleção encadeada;

Inicio

Inteiro a, b, c;

Leia(a, b, c);

Se ( (a > b) e (a > c) e (b > c) ) então

Escreva("A sequência é: ", a,", ", b," e ", c,".");

senão

se ( (a > b) e (a > c) e (c > b) ) então

Escreva("A sequência é: ", a,", ", c," e ", b,".");

fimse

fimse

Se ( (b > a) e (b > c) e (a > c) ) então

Escreva("A sequência é: ", b,", ", a," e ", c,".");

senão

se ( (b > a) e (b > c) e (c > a) ) então

Escreva("A sequência é: ", b,", ", c," e ", a,".");

fimse

fimse

Se ( (c > b) e (c > a) e (b > a) ) então

Escreva("A sequência é: ", c,", ", b," e ", a,".");

senão

se ( (c > b) e (c > a) e (a > b) ) então

Escreva("A sequência é: ", c,", ", a," e ", b,".");

fimse

fimse

Fim

1. Faça um algoritmo que leia o ano de nascimento de uma pessoa, calcule e mostre sua idade e, também, verifique se mostre se ela já tem idade para votar (16 anos ou mais) e para conseguir a Carteira de habilitação (18 anos ou mais).

Inicio

inteiro ano, idade;

Leia(ano);

idade = 2021 - ano;

Escreva("Sua idade é: ",idade);

se (idade >= 18) então

Escreva("Você já pode dirigir");

Escreva("Você já pode votar");

senão

se (idade >=16) então

Escreva("Você já pode votar");

fimse

fimse

fim

**Seleção de múltipla escolha**

Exemplo 1

escolha (x)

início

caso 1:

comando 1;

caso 2:

comando 2;

 caso 3, 4:

comando 3\_4;

caso 5..10:

comando 5 até 10;

caso contrário:

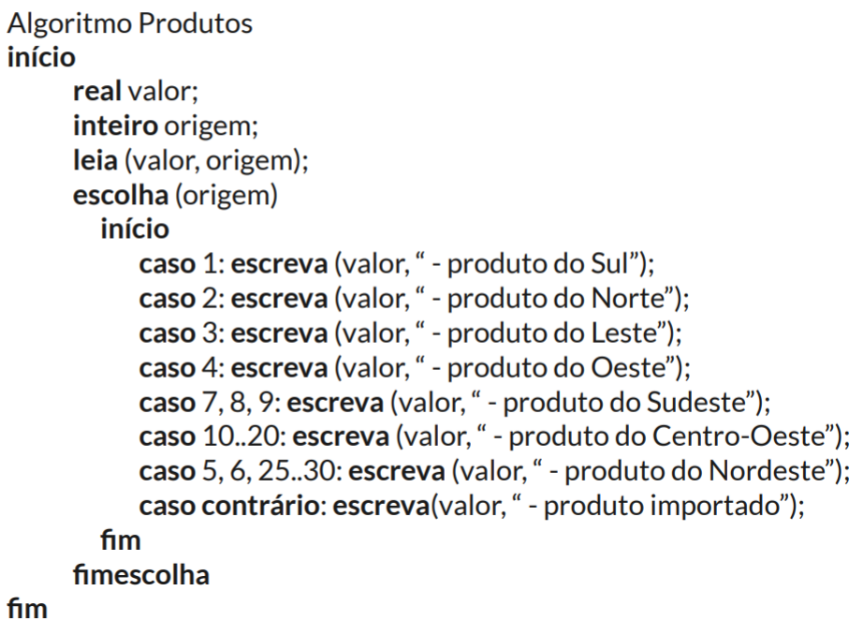
comando caso contrário;

fim

fim escolha

Exemplo 2

Construa um algoritmo que, tendo como dados de entrada o valor de um produto e seu código de origem, mostre o preço junto de sua procedência. Caso o código não seja nenhum dos especificados, o produto deve ser encarado como importado. Siga a tabela de códigos a seguir:



**Exercício avaliado**

|  |  |
| --- | --- |
| Código | Classificação |
| 1 | Alimentos não-perecíveis |
| 2, 3 ou 4 | Alimentos Perecíveis |
| 5 ou 6 | Vestuário |
| 7 | Higiene pessoal |
| 8 até 15 | Limpeza e utensílios domésticos |
| Outro código | Inválido |

Escreva um algoritmo que leia o código de um determinado produto e mostre a sua classificação. Utilize a seguinte tabela como referência:

Inicio

Inteiro cod;

Caracter texto;

texto = "A classificação deste produto é"

Leia(cod);

Escolha(cod)

inicio

caso 1: escreva (texto,": Alimentos não pereciveis");

caso 2, 3, 4: escreva (texto,": Alimentos pereciveis");

caso 5, 6: escreva (texto,": Vestuário");

caso 7: escreva (texto,": Higiene pessoal");

caso 8..15: escreva (texto,": Limpeza e utensílios domésticos");

caso contrario: escreva (texto,": Inválido");

fim

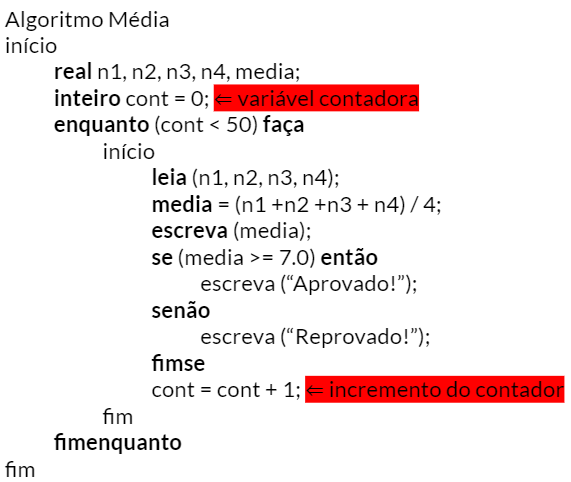
fimescolha

fim

****

**Aula 5 - Estrutura de Repetição**

* **Estrutura de repetição com teste no início (enquanto).**

Ex

enquanto (<condição>) faça

início

comando 1;

comando 2;

comando 3;

fim

fimenquanto

**Variável contadora**

É um tipo de variável utilizada para “contar” um determinado valor a cada instante de tempo t.

**Variável acumuladora**

É um tipo de variável utilizada para “acumular” valores. Por exemplo, um somatório é um exemplo de uma variável acumuladora.

**Exercícios**

1. Faça um algoritmo que escreva os números ímpares entre 1 e 1000.

Inicio

inteiro cont = 1;

enquanto (cont <= 1000) faça

inicio

se (cont % 2 == 1) então

escreva(cont);

fimse

cont = cont + 1;

fim

fimenquanto

fim

1. Faça um algoritmo que calcule o somatório de todos os números pares entre dois números informados pelo usuário.

Inicio

inteiro som = 0, ini, fim;

leia(ini, fim);

enquanto (ini<=fim) faça

inicio

se (ini % 2 == 0) então

som = som + ini;

fimse

ini = ini + 1;

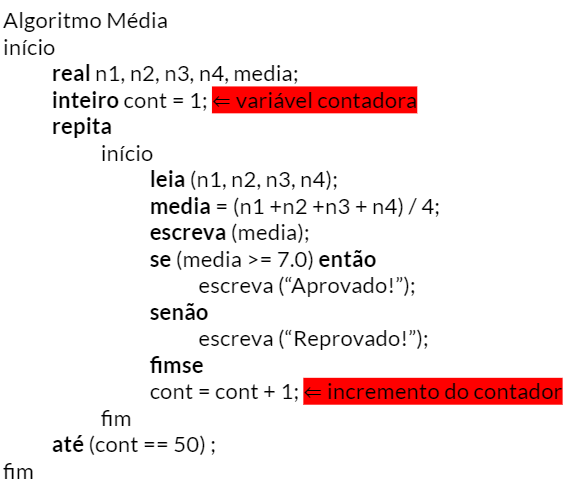
fim

fimenquanto

escreva(som);

fim

* **Estrutura de repetição com teste no final**

Nesta estrutura de repetição, o bloco de comandos é executado pelo menos uma vez, independente da validade da condição. Isto ocorre porque a inspeção da condição é feita após a execução do bloco.

Ex:

repita

início

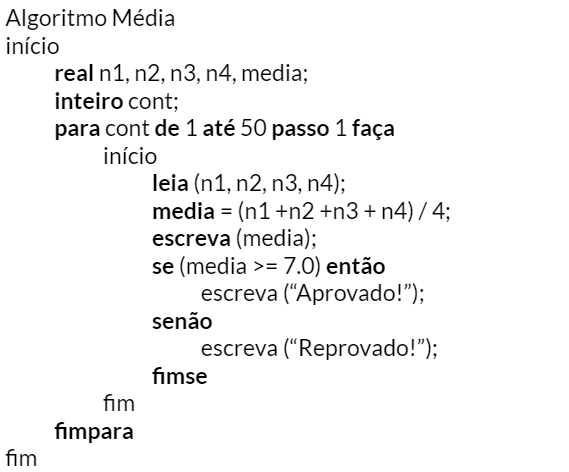
comando 1;

comando 2;

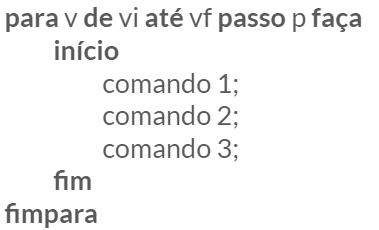
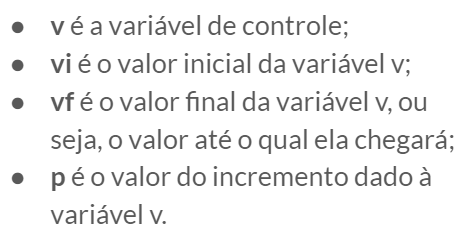
comando 3;

fim

até (<condição>);



* **Estrutura de repetição pare-faça**



Guilherme Pereira do Amarilho

EXERCÍCIOS AVALIADOS

QUESTÃO 5

Inicio

inteiro num, cont = 1, aux;

real soma = 1, fat;

leia(num);

enquanto (cont <= num) faça

inicio

aux = 1;

fat = 1;

enquanto (aux <= cont) faça

inicio

fat = fat \* aux;

aux = aux + 1;

fim

fimenquanto

soma = soma + (1/fat);

cont = cont + 1;

fim

fimenquanto

fim

QUESTÃO 8

inicio

inteiro resposta = 1, cand1, cand2, cand3, cand4, nulo, branco;

enquanto (resposta != 0) faça

inicio

leia(resposta);

escolha(resposta)

inicio

caso 1: cand1 = cand1 + 1;

caso 2: cand2 = cand2 + 1;

caso 3: cand3 = cand3 + 1;

caso 4: cand4 = cand4 + 1;

caso 5: nulo = nulo + 1;

caso 6: branco = branco + 1;

fim

fimescolha

fim

fimenquanto

escreva("O candidato 1 recebeu: ",cand1," votos");

escreva("O candidato 2 recebeu: ",cand2," votos");

escreva("O candidato 3 recebeu: ",cand3," votos");

escreva("O candidato 4 recebeu: ",cand4," votos");

escreva("Houve: ",nulo," votos nulos");

escreva("Houve: ",branco," votos em branco");

fim

QUESTÃO 15

inicio

real doism = 0, maior = 0, media = 0, altura;

inteiro cont = 0;

enquanto (cont < 20) faca

inicio

leia(altura);

se (altura > maior) então

maior = altura;

fimse

se(altura >= 2) então

doism = doism +1;

fimse

media = media + altura;

cont = cont + 1;

fim

fimenquanto

media = media / 20;

escreva(maior);

escreva(media);

escreva(doism);

fim

QUESTÃO 18

inicio

inteiro idade, sexo, experiencia, fem, masc, ho45, mu35, nova;

real medHomem;

enquanto (idade != 0) faca

inicio

leia(idade,sexo, experiencia);

se (idade !=0) então

se (sexo == 1) então

masc = masc + 1;

senao

fem = fem + 1;

fimse

se (sexo == 1 e experiencia == 1) então

medHomem = medHomem + idade;

fimse

se (sexo == 1 e idade >= 45) então

ho45 = ho45 + 1;

fimse

se (sexo == 2 e experiencia == 1 e idade < 35) então

mu35 = mu35 + 1;

fimse

se (nova == 0 e sexo == 2) então

nova = idade;

fimse

se (idade < nova e sexo == 2 e experiencia == 1) entao

nova = idade;

fimse

fimse

fim

fimenquanto

escreva("há ",fem," mulheres.");

escreva("há ",masc," homens.");

medHomem = medHomem/masc;

escreva("A idade média dos homens com experiencia de trabalho é: ",medHomem," anos");

ho45 = ho45/masc\*100;

escreva("A porcentagem dos homens com mais de 45 anos entre o total dos homens é: ",ho45,%");

escreva("número de mulheres com idade inferior a 35 anos e com experiência no serviço é de: ",mu35);

se (nova == 0) então

escreva("Não houve mulheres cadastradas")

senao

escreva("A idade da mulher mais nova com experiencia é: ",nova," anos.");

fimse

fim

QUESTÃO 19

inicio

inteiro nota1, nota2, contador = 0, aprovados = 0, reprovados = 0, recuperacao = 0;

real media, mediageral;

enquanto(contador < 6) faça

inicio

leia(nota1,nota2);

media = (nota1 + nota2)/2;

mediageral = mediageral + media;

imprima("Sua média é: ",media);

se (media>=7) então

aprovados = aprovados + 1;

imprima("Você foi aprovado!");

senão

se (media > 5) então

imprima("Você ficou em recuperação!");

recuperacao = recuperacao + 1;

senão

imprima("Você Reprovou!");

reprovados = reprovados + 1;

fimse

fimse

contador = contador + 1;

fim

fimenquanto

mediageral = mediageral/6

imprima("O total de alunos aprovado foi de: ",aprovado);

imprima("O total de alunos reprovados foi de: ",reprovados);

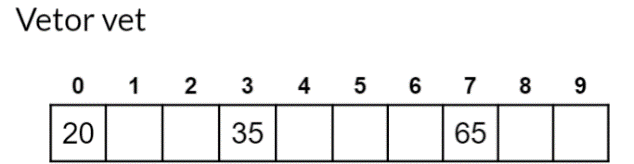
imprima("O total de alunos em recuperação foi de: ",recuperacao);

imprima("A média geral de todos os alunos é: ",mediageral);

fim

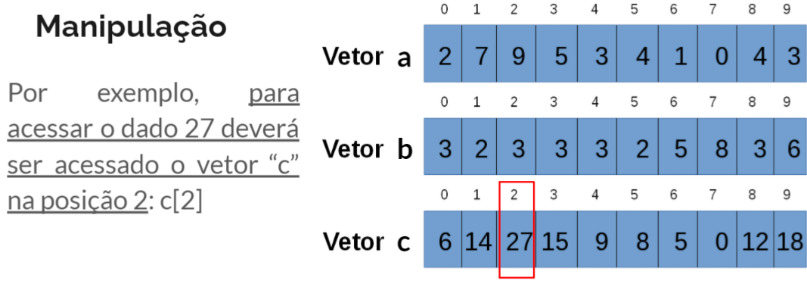
**ESTRUTURA DE DADOS**

VETORES

* Variáveis compostas unidimensionais

Ex:

Int vet[10]



* Manipulação

A manipulação dos vetores é realizada pelo seu índice. Cada “variável” é independente e pode ser acessada por um índice que no caso da linguagem de programação C, inicia em zero.

* Utilização

c[1] = 20; //Para atribuir o valor 20 no vetor “c” na posição 1

scanf(“%i”, &a[5]); // Para ler um valor para o vetor “a” na posição 5

printf(“%i”, b[6]); // Para escrever o valor correspondente do vetor “b” na posição 6

Ex:

#define TAM 50

int vet[TAM], j;

//preenchendo o vetor

for (j=0; j < TAM; j++){

printf("Digite um valor para vet[%i]: ", j);

scanf("%i", &vet[j]);

}

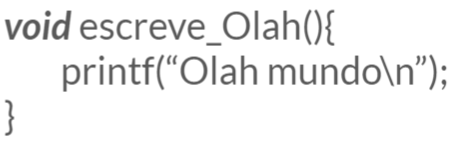
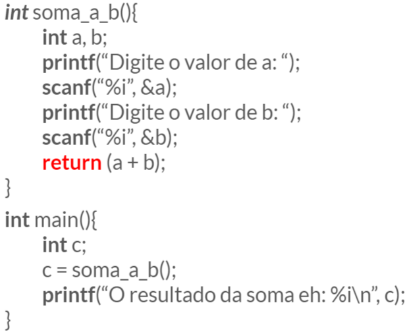
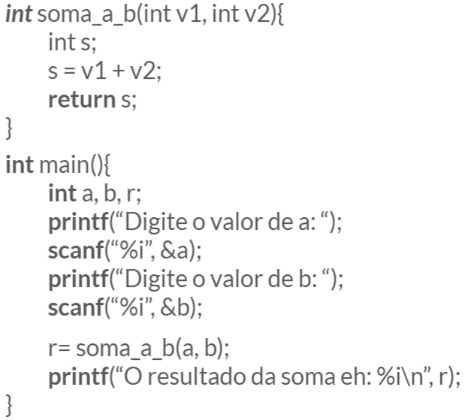
//lendo o vetor

for (j=0; j < TAM; j++){

printf("%i", vet[j]);

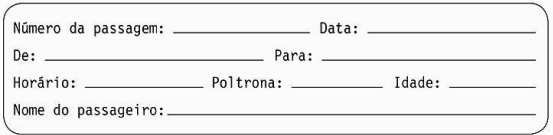
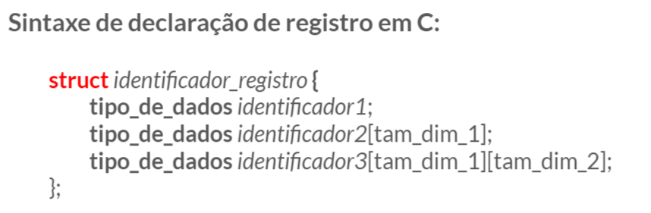
}

**MODULARIZAÇÃO**

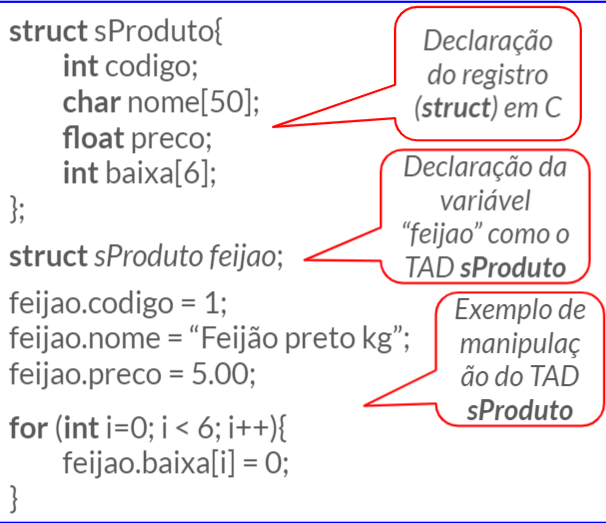
* Decomposição de problemas
  + Fator determinante para a redução da complexidade;
  + Ajuda a simplificar a resolução;
  + Permite focar a atenção em um problema pequeno por vez;
  + Facilita a compreensão do todo;
* Módulo de ação (Procedimento)
  + Todo procedimento possui um tipo de dado que é vazio (void);
  + Não há retorno.
* Módulo de resultado (Função)
  + Retornam um valor especifico
  + Toda função deve ter um tipo de dado o qual determina o tipo de dado do retorno;
  + A expressão contida dentro do comando return é chamada de valor de retorno (é a resposta da função);
  + O comando return é sempre executado por último.
* Passagem de parâmetros
  + Um módulo pode não ter parâmetros:
  + Parâmetros são utilizados para enviar informações para dentro do módulo;

**REGISTRO**

O registro é uma variável composta (possui elementos que são partes que especificam cada uma das informações do todo) e heterogênea (cada elemento pode ser de um tipo primitivo de dados diferente).



Tipo Inteiro: Número da passagem, poltrona.

Tipo Caractere: De, Para, Nome do passageiro.

