

PROGRAMADOR

ALGORITMOS E LÓGICA
DE PROGRAMAÇÃO



Introdução à
lógica

1

Modelo utilizado
no curso

2

Comentário

3

Tipos de dados

4

Estrutura de
controle

5

Estrutura de
repetição

6

Vetor

7

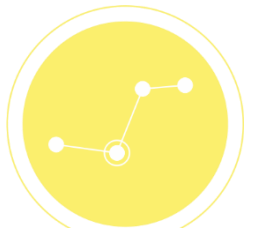
Matriz

8

Programação
estruturada
ou modular

9

CONTEÚDO DO MÓDULO





Conteúdo Aula 1

Introdução à
lógica

1

Linguagem
Natural

2

Fluxograma

3

Português
Estruturado

4



INTRODUÇÃO A LÓGICA

Todos temos que utilizar a lógica, principalmente profissionais de TI;
Não iremos aprender a pensar, mas sim desenvolver e aperfeiçoar
melhor esta técnica;

Lembre-se para isso você deve ser persistente praticar constantemente;
Isso fará com que você aprenda cada vez mais;

Para aplicarmos a lógica vamos aprender um pouco sobre **algoritmos**;

O que é **Algoritmo**?

Quando ouvimos a primeira vez algoritmo logo pensamos que
sabemos o que é.

Algoritmo lembra uma palavra que estamos acostumados a ouvir desde
criança: **Algarismo**.

São coisas totalmente diferente Algoritmo e Algarismo;



INTRODUÇÃO A LÓGICA

Os algoritmos são escritos de maneira precisa, em linguagem quase informal.

Algoritmos são conceitos simples utilizados por nós diariamente;
Diariamente utilizamos algoritmos digitais;

Acessamos sites de busca;

Lemos e-mails;

Realizamos compras;

Acessamos redes sociais;

Todos os aplicativos que utilizamos em nossos celulares;

Aplicativos em sua SmartTv;

Todos esses sistemas existem por conta dos algoritmos;



INTRODUÇÃO A LÓGICA

Algoritmos são conjuntos de passos **finitos** e **organizados** que, quando executados, resolvem um determinado **problema**;

Para resolver um problema através de um computador é necessário encontrar em primeiro lugar uma maneira de descrevê-lo de uma forma clara e precisa.

É necessário também que seja encontrada uma **sequência de passos** que leve a solução do problema.

A necessidade de uma rede social fez surgir o Facebook;

Em nosso dia a dia, quando resolvemos fazer um bolo e seguimos uma receita, estamos utilizando de um algoritmo a receita;



INTRODUÇÃO A LÓGICA

Uma outra rotina no nosso cotidiano: **Atravessar a rua.**

Algoritmo AtravessarRua

Olhar para a direita

Olhar para a esquerda

Se estiver vindo carro

Não atravesse

Senão

Atravesse

Fim-se

Fim-Algoritmo



INTRODUÇÃO A LÓGICA

Problemas relacionados a lógica são os erros mais difíceis de localizar. Esses erros não são apresentados durante a criação, mas sim durante a execução.

O que pode nos causar uma dor de cabeça para descobrir onde encontra-se o erro.



INTRODUÇÃO A LÓGICA

O algoritmo a seguir está correto? Por quê?

Algoritmo AtravessarRua

Olhar para a esquerda

Olhar para a direita

Se não estiver vindo carro

 Não atravesse

senão

 Atravesse

Fim-se

Fim-Algoritmo



INTRODUÇÃO A LÓGICA

Passos na concepção do algoritmo:

- Analisar o problema;

- Identificar os dados de entrada;

- Identificar os dados de saída;

- Verificar o que é preciso para transformar a entrada no dado de saída;

 - Utilização da estratégia dividir para conquistar;

 - Analisar as regras e restrições necessárias;

 - Realizar os processamentos necessários;

 - Garantir ausência de ambiguidade;

- Construir o algoritmo;

- Testar;

- Executar;



INTRODUÇÃO A LÓGICA

Existem diversos tipos de algoritmos diferentes.

Vamos aprender os seguintes:

- Linguagem Natural;

- Fluxograma;

- Português Estruturado;



LINGUAGEM NATURAL

Os algoritmos são escritos em nossa linguagem natural (Português):

Entrada: ingredientes;

Saída: bolo;

Algoritmo: receita;

Bata as claras em neve e reserve;

Misture as gemas, a margarina e o açúcar até obter uma massa homogênea;

Acrescente o leite e a farinha de trigo aos poucos sem parar de bater;

Por último, adicione as claras em neve e o fermento;

Despeje a massa em uma forma grande de furo central untada e enfarinhada;

Asse em forno médio (180° C), preaquecido, por 40 minutos ou ao furar com um garfo e este saia limpo;



LINGUAGEM NATURAL

Problema:

Somar 2 valores;

Entradas:

2 valores reais, num1 e num2;

Saídas:

A soma: $soma = num1 + num2$;

Algoritmo:

Início

Ler num1, num2

Calcular a soma de num1 e num 2

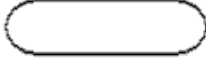


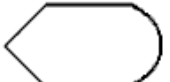



Escrever soma

Fim



FLUXOGRAMA

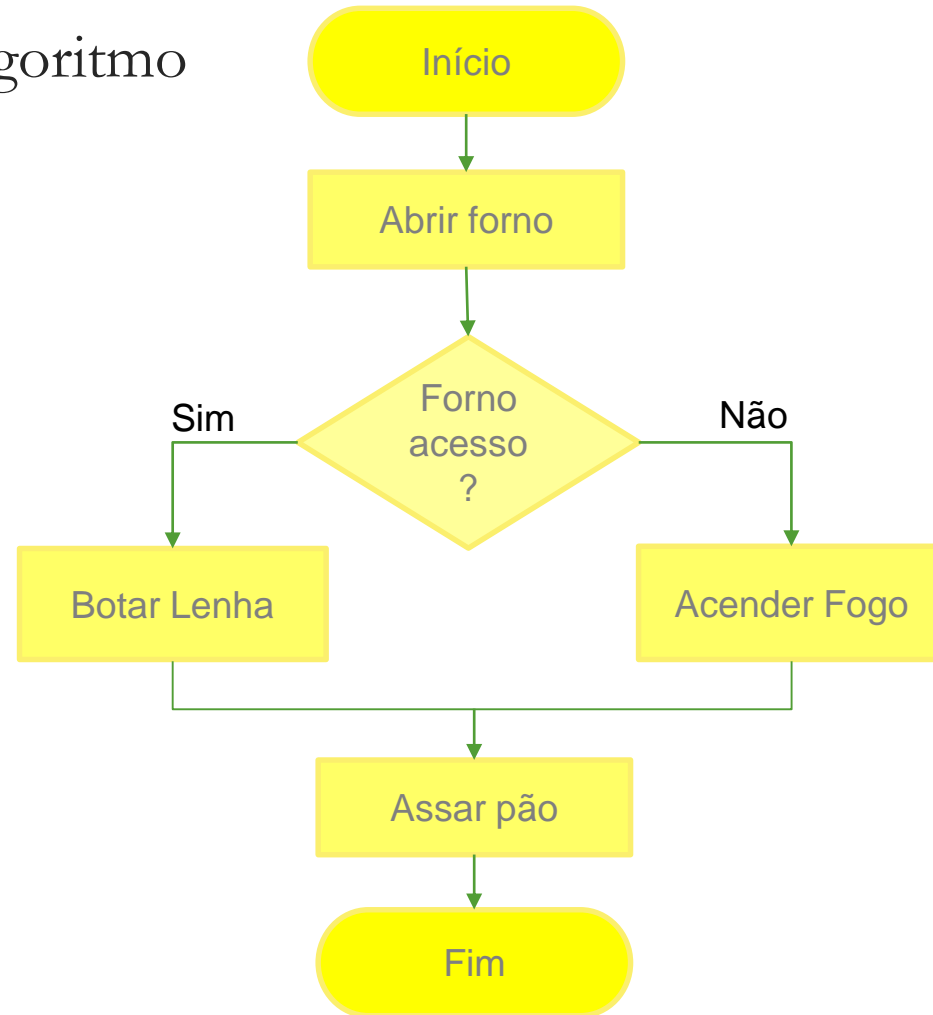
Representação gráfica que emprega formas geométricas padronizadas para indicar as diversas ações e decisões que devem ser executadas para resolver o problema.

| FIGURA | SIGNIFICADO |
|---|---|
|  | Figura para definir início e fim do algoritmo |
|  | Figura usada no processamento de cálculo, atribuições e processamento de dados em geral |
|  | Figura utilizada na representação de entrada de dados |
|  | Figura utilizada para representação da saída de dados |
|  | Figura que indica o processo seletivo ou condicional, possibilitando o desvio no caminho do processamento |
|  | Símbolo geométrico usado como conector |
|  | Símbolo que identifica o sentido do fluxo de dados, permitindo a conexão entre as outras figuras existentes |



FLUXOGRAMA

Exemplo de um algoritmo em fluxograma.





PORTUGUÊS ESTRUTURADO

Utiliza uma linguagem intermediária entre a linguagem natural e a linguagem de programação para realizarmos a representação de algoritmos.

início

<instruções>

se <teste> então

<instruções>

senão

<instruções>

fim_se

fim;



PORTUGUÊS ESTRUTURADO

início

var

num1, num2, resultado: inteiro

escreva(“Digite um valor para o primeiro número: ”)

leia (num1)

escreva(“Digite um valor para o segundo número: ”)

leia (num2)

resultado num1 + num2

escrever(resultado)

fim;



EXERCÍCIOS

Façam a representação dos exercícios abaixo nos modelos: linguagem natural, fluxograma e português estruturado.

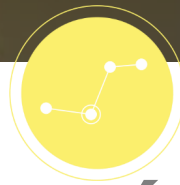
1. Escreva um algoritmo para armazenar o valor 20 em uma variável X e o valor 5 em uma variável Y. A seguir, armazenar a soma do valor de X com o de Y em uma variável Z. Escrever (na tela) o valor armazenado em X, em Y e em Z.
2. Escreva um algoritmo para armazenar o valor 4 em uma variável A e o valor 3 em uma variável B. A seguir, armazenar a soma de A com B em uma variável C e a subtração de A com B em uma variável D. Escrever o valor de A, B, C e D e também escrever a mensagem 'Fim do Algoritmo'.



EXERCÍCIOS

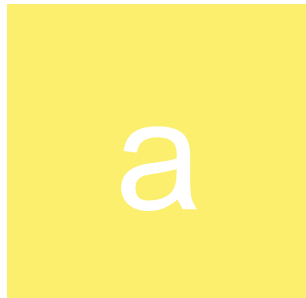
Façam a representação dos exercícios abaixo nos modelos: linguagem natural, fluxograma e português estruturado.

3. Criar um algoritmo que leia a idade de 5 pessoas e apresente a idade média delas.

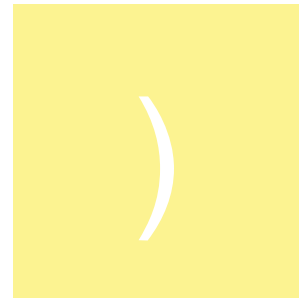


EXERCÍCIOS

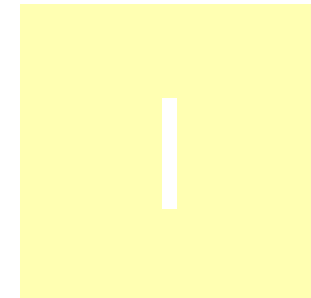
Qual o modelo acharam o mais fácil?



Linguagem
natural?



Fluxograma?



Português
estruturado



Conteúdo Aula 2

Modelo utilizado
no curso

1

Visualg

2

Comentário

3

Tipo de dados

4



MODELO UTILIZADO NO CURSO

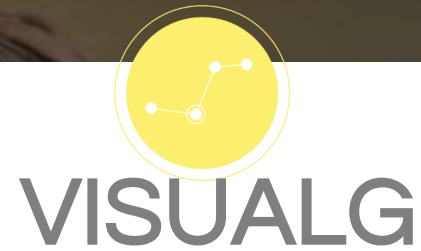
Não existe um consenso sobre qual a melhor maneira de representar um algoritmo.

Adotaremos o português estruturado para representar nossos algoritmos.

Essa forma de representação tem a vantagem de o algoritmo ser escrito de uma forma que está próxima de uma linguagem de programação.

Para já os familiarizarmos com alguma ferramenta de desenvolvimento, vamos utilizar o visualg:

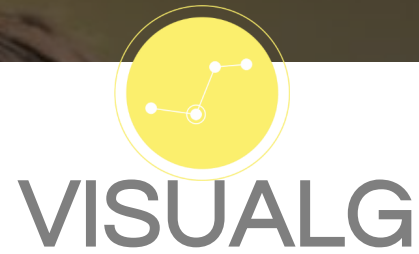
<http://www.apoioinformatica.inf.br/produtos/visualg>



Abrir o programa visualg que instalaram anteriormente.
Digitar o algoritmo abaixo:

```
algoritmo "OlaMundo"  
inicio  
  Escreva("Olá Mundo!")  
  Escreva("Tudo na mesma linha com o Escreva.")  
  Escreval("Com o Escreval pula linha.")  
  Escreval("Nova linha.")  
finalgoritmo
```

Após digitar todo o algoritmo como o exemplo acima, pressione a tecla F9 que é a tecla responsável por executar o algoritmo.



O que foi apresentado no console do programa?

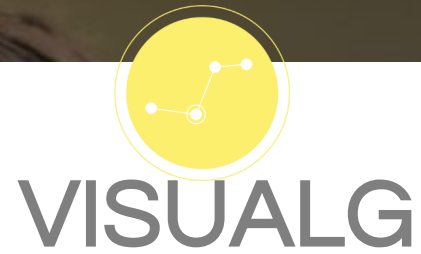
Se foi digitado tudo correto, e não ocorreu nenhum erro o resultado obtido foi igual as mensagens abaixo:

"Olá Mundo!Tudo na mesma linha com o Escreva.Com o Escreval pula linha."

“Nova linha.”

O comando **Escreva**, escreve o que você mandarem ele escrever no console.

O comando **Escreval**, escreve o que você mandarem ele escrever no console e após escrever pula uma linha.



```
algoritmo "OlaMundo"  
var  
nome : caractere  
inicio  
Escreva("Digite o seu nome: ")  
leia(nome)  
Escreval(nome)  
finalgoritmo
```

Agora aprendemos como solicitar que o usuário digite informações pelo teclado.



IDENTAÇÃO

```
algoritmo "OlaMundo"
```

```
inicio
```

```
    Escreva("Olá Mundo!")
```

```
    Escreva("Tudo na mesma linha com o Escreva.")
```

```
    Escreval("Com o Escreval pula linha.")
```

```
    Escreval("Nova linha.")
```

```
fimalgoritmo
```

O código como apresentado acima fica mais fácil de se entender.

Isso se chama indentação de código.

É sempre bom implementarmos nossos códigos com indentação, para termos mais clareza ao analisarmos o código.

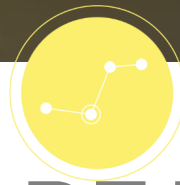


COMENTÁRIO

Comentários são muito importantes em seus programas. Eles são utilizados quando vocês desejam deixar algo comentado no código para que outras pessoas vejam no futuro, ou mesmo vocês.

Também é utilizando quando desejamos comentar temporariamente alguma parte do código.

//Escreva(“Esta linha estará comentada, portanto não será executada.”)



TIPOS DE DADOS

algoritmo "TiposDados"

var

nome : caractere

idade : inteiro

salario : real

masculino : logico

inicio

nome <- "Maria"

idade <- 18

salario <- 100,00

masculino <- falso

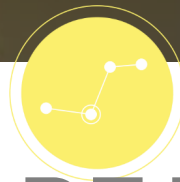
escreval("Nome: ", nome)

escreval("Idade: ", idade)

escreval("Salario: ", salario)

escreval("Masculino?: ", masculino)

fimalgoritmo



TIPOS DE DADOS

Todo programa deve permitir de alguma forma a entrada e saída de dados do mundo exterior.

Os **dados** são elementos do mundo exterior representados no computador.

Eles podem ser classificados em 3 tipos:

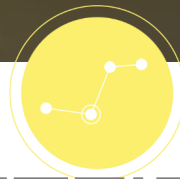
- Primitivos;

- Básicos:

 - Numéricos (valores inteiros e reais);

 - Caracteres (valores alfabéticos ou alfanuméricos);

 - Lógicos (valores dos tipos falsos ou verdadeiros);



INTEIRO

São todos os valores inteiros positivos e negativos.

28

87

2016

-48

-127

Iremos fazer referência a esse tipo de dado como inteiro nos algoritmos.

Esse tipo de dados é utilizado principalmente em operações de cálculos.



REAL

São todos os valores positivos e negativos tanto os inteiros quanto os fracionados.

28

87.5

-48

-127.98

Iremos fazer referência a esse tipo de dado como real nos algoritmos. Esse tipo de dados é utilizado principalmente em operações de cálculos.



CARACTERE

São os caracteres delimitados pelo símbolo “”.

São representados por:

- Letras (de A até Z);

- Números (de 0 até 9);

- Símbolos (todos os símbolos imprimíveis existentes no teclado);

- “Programação”;

- “Avenida Jornalista Rubens de Arruda Ramos, Beira Mar, 32 - Apto. 1”;

Iremos fazer referência a esse tipo de dado como caractere nos algoritmos.

Esse tipo de dados são geralmente utilizados em entradas e saídas dedados no sentido de obter ou apresentar uma mensagem.



LÓGICO

São os dados com valores sim ou não, verdadeiro ou falso, 1 ou 0.

Apenas 1 dos valores pode ser escolhido para ser armazenado.

São representados por:

verdadeiro

falso

Iremos fazer referência a esse tipo de dado como logico nos algoritmos.

Esse tipo de dados é utilizado principalmente em operações de cálculos.



EXERCÍCIO

1. O que acontece se for incluído um valor real em uma variável inteira?
2. Qual seria o tipo de variável correto para armazenar os seguintes dados:
 - a. A sua idade.
 - b. A área da sua casa.
 - c. O número de estrelas no céu.
 - d. O endereço da sua casa.
3. Faça um algoritmo declarando 4 variáveis de tipos diferentes.
4. Obtendo 2 notas do teclado, calcule a média delas e apresente a média na tela.



EXERCÍCIO

5. Obtenha o nome, telefone, endereço e idade de uma pessoa. Após armazenar os dados nas variáveis, apresente os dados na tela.
6. Calcule o valor de cada expressão abaixo e indique o tipo do resultado (inteiro ou real):
- a. $(20 - 15)/2$ b)
 - b. $20 - 15/2$ c)
 - c. $2*5/20 + 30/15*2$
 - d. $2*(5/20) + 30/(15*2)$
 - e. $23 \text{ div } 4$
 - f. $23 \text{ mod } 4$
 - g. $35 \text{ div } 6 + 2$
 - h. $35 \text{ div } 6 - 2$



EXERCÍCIO

7. Qual é o resultado das operações quando o algoritmo encontrar as seguintes linhas:
- a. Escreva ('mario' = 'maria');
 - b. Escreva ($2 + 4 = 6$);
 - c. Escreva ($10 - 4 > 7$);
 - d. Escreva ($((2*3) > (3*2))$);
 - e. Escreva (não('a' > 'A'));
8. Faça um algoritmo que calcule a área de um triângulo, considerando a fórmula $area \leftarrow (base * altura) \div 2$. Utilize as variáveis area, base e altura e os operadores aritméticos de multiplicação e divisão.



EXERCÍCIO

9. Faça um algoritmo que leia uma temperatura em graus Celsius e apresente-a convertida em graus Fahrenheit. A fórmula de conversão é: $f \leftarrow (9 * c + 160) \text{ div } 5$, na qual f é a temperatura em Fahrenheit e c é a temperatura em Celsius.
10. Faça um algoritmo que leia dois valores para as variáveis A e B e efetue a troca dos valores de forma que a variável A passe a possuir o valor da variável B e a variável B passe a possuir o valor da variável A . Apresente os valores trocados.
11. Faça um algoritmo que leia a velocidade de um veículo em km/h e calcule e imprima a velocidade em m/s (metros por segundo).



EXERCÍCIO

12. Faça um algoritmo que calcule a quantidade de litros de combustível gasta em uma viagem, utilizando um automóvel que faz 12Km por litro. Para obter o cálculo, o usuário deve fornecer o tempo gasto na viagem e a velocidade média durante ela. Desta forma, será possível obter a distância percorrida com a fórmula $distancia \leftarrow tempo * velocidade$. Tendo o valor da distância, basta calcular a quantidade de litros de combustível utilizada na viagem com a fórmula: $litros = distancia \div 12$. O programa deve apresentar os valores da velocidade média, tempo gasto na viagem, a distância percorrida e a quantidade de litros utilizada na viagem.



Conteúdo Aula 3

Estrutura de Controle

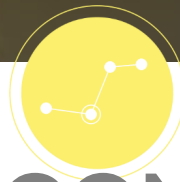
1

Desvio condicional
Simples

a.

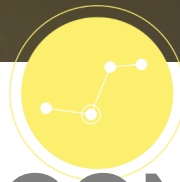
Desvio condicional
composto

b.



ESTRUTURA DE CONTROLE

```
algoritmo "EstruturaControle"  
var  
    num1, num2 : inteiro  
inicio  
    num1 <- 1  
    num2 <- 2  
    se num1 > num2 entao  
        escreval("Número 1 é maior do que número 2")  
    senao  
        escreval("Número 2 é maior do que número 1")  
    fimse  
finalgoritmo
```

ESTRUTURA DE CONTROLE

Se tivermos 2 caminhos a seguir no código. O que fazer?

Quando vamos atravessar uma rua precisamos tomar uma decisão.

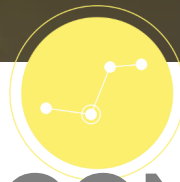
Ou será que podemos sempre atravessar a rua e confiar que nunca estará vindo um carro?

É necessário entendermos o que significa **condição** e **decisão**.

Condição: algo imposto e aceito ou não.

Decisão: efeito de uma tomada de decisão com base na condição imposta.

Essa estrutura de controle é também conhecida como **controle de fluxo** ou **desvio condicional**.



ESTRUTURA DE CONTROLE

Há vários tipos de desvios condicionais:

- Desvio condicional simples.

- Desvio condicional composto.

- Desvio condicional encadeado.

- Tomada de decisão por seleção.



DESVIO CONDICIONAL SIMPLES

```
algoritmo "CondicionalSimples"  
var  
    num1, num2 : inteiro  
inicio  
    num1 <- 1  
    num2 <- 2  
    se num1 > num2 entao  
        escreval("Número 1 é maior do que número 2")  
    fimse  
finalgoritmo
```



DESVIO CONDICIONAL SIMPLES

Nesse exemplo, podemos ver que caso a condição seja verdadeira, é executado o(s) trecho(s) de código que estiverem entre os comandos **se...então** e **fimse**.

Após a execução ou não desse bloco de instruções, são executadas as eventuais instruções que existem após o comando **fimse**.



DESVIO CONDICIONAL COMPOSTO

```
algoritmo "CondicionalComposto"  
var  
    num1, num2 : inteiro  
inicio  
    num1 <- 1  
    num2 <- 2  
    se num1 > num2 entao  
        escreval("Número 1 é maior do que número 2")  
    senao  
        escreval("Número 2 é maior do que número 1")  
    fimse  
finalgoritmo
```



DESVIO CONDICIONAL COMPOSTO

Nesse exemplo, podemos ver que caso a condição seja verdadeira, é executado o(s) trecho(s) de código que estiverem entre os comandos **se...então** e **senão**. Caso a condição seja falsa, é executado o(s) trecho(s) de código que estiverem entre os comandos **senão** e **fimse**. Após a execução ou não desse bloco de instruções, são executadas as eventuais instruções que existem após o comando **fimse**.



EXERCÍCIO

1. Escrever um algoritmo que leia o nome e as três notas obtidas por um aluno durante o semestre. Calcular a sua média (aritmética), informar o nome e sua menção aprovado (media ≥ 7), Reprovado (media < 7).
2. Escrever um algoritmo que leia dois valores inteiros distintos e informe qual é o maior.
3. Elabore um algoritmo que leia dois números inteiros e mostre o resultado da diferença do maior valor pelo menor.
4. Elabore um algoritmo que leia 5 valores inteiros e apresente na tela o maior e o menor deles.
5. Faça um algoritmo que leia um número e mostre uma mensagem indicando se este número é par ou ímpar e se é positivo ou negativo.



EXERCÍCIO

6. Escrever um algoritmo que lê um valor em reais e calcule qual o menor número possível de notas de 100, 50, 10, 5 e 1 em que o valor lido pode ser decomposto. Escrever o valor lido e a relação de notas necessárias.
7. Num determinado Estado, para transferências de veículos, o DETRAN cobra uma taxa de 1% para carros fabricados antes de 1990 e uma taxa de 1.5% para os fabricados de 1990 em diante, taxa esta incidindo sobre o valor de tabela do carro. O algoritmo abaixo lê o ano e o preço do carro e a seguir calcula e imprime imposto a ser pago.



Conteúdo Aula 4

Estrutura de Controle

1

Desvio condicional
encadeado

a.

Tomada de
decisão por
seleção

b.

Estrutura de
repetição

2

Repetição com
interrupção
início

a.



DESVIO CONDICIONAL ENCADEADO

```
algoritmo "CondicionalEncadeado"  
var  
    num1, num2 : inteiro  
inicio  
    num1 <- 1  
    num2 <- 2  
    se num1 > num2 entao  
        escreval("Número 1 é maior do que número 2")  
    senao  
        escreval("Número 2 é maior do que número 1")  
        se num2 MOD 2 = 0 entao  
            escreval("Número 2 é um número par.")  
        fimse  
    fimse  
finalgoritmo
```

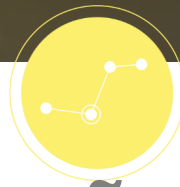


DESVIO CONDICIONAL ENCADEADO

Nesse caso, a diferença é que podemos acrescentar quantas condicionais quisermos dentro de outra condicional.

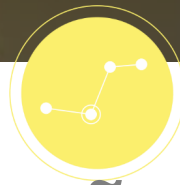
Podemos combinar várias condicionais, gerando uma gama maior de possibilidades.

Dessa forma podemos ter tomadas de decisão encadeadas utilizando tomada de decisão simples em conjunto com tomadas de decisão composta.



TOMADA DE DECISÃO POR SELEÇÃO

```
algoritmo "DecisaoSelecao"  
var  
  nota : inteiro  
inicio  
  nota <- 7  
  escolha nota  
  caso 0,1,2,3  
    escreval("Reprovado.")  
  caso 5 ate 7, 4  
    escreval("Em recuperação.")  
  caso 8 ate 10  
    escreval("Aprovado")  
  outrocaso  
    escreval("Nota inválida.")  
  fimescolha  
finalgoritmo
```

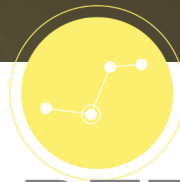



TOMADA DE DECISÃO POR SELEÇÃO

Colocamos várias opções e vários comandos dentro do comando escolha, todas as possibilidades de rumo que nossos programas possam tomar.

A escolha vai funcionar como um interruptor, pois dependendo da entrada que você der a ele, ele vai acionar somente certo(s) comando(s) dentre os que você disponibilizou.

É como se você criasse um menu, ou cardápio, e com o escolha você escolhesse o que vai querer.



ESTRUTURA DE REPETIÇÃO

O que devemos fazer para imprimir uma sequência numérica, por exemplo de 1 até 100?

Escreva("1")

Escreva("2")

Escreva("3")

Escreva("4")

Escreva("5")

Escreva("6")

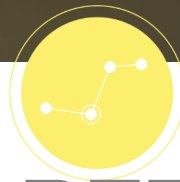
Escreva("7")

Escreva("8")

Escreva("9")

...

Isto está correto?



ESTRUTURA DE REPETIÇÃO

Quanto tempo iremos perder fazendo o algoritmo dessa forma?

E quanto código duplicado teremos?

Quando precisarmos realizar uma repetição de um código, podemos utilizar as estruturas de repetição:

- Repetição com interrupção início;

- Repetição com interrupção no fim;

- Repetição com variável de controle;



ESTRUTURA DE REPETIÇÃO COM INTERRUPÇÃO NO INÍCIO

```
algoritmo "EstruturaRepeticao"  
var  
    count : inteiro  
inicio  
    enquanto count <= 100 faca  
        Escreval(count)  
        count <- count + 1  
    fimenquanto  
finalgoritmo
```



ESTRUTURA DE REPETIÇÃO COM INTERRUPÇÃO NO INÍCIO

Enquanto o resultado da condição for verdadeiro irá continuar executando o código entre os comandos **faca...fimenquanto**.



EXERCÍCIO

1. Escrever um algoritmo que leia o nome e as 4 notas do semestre de um aluno. Calcular a média, informar o nome e:

Aprovado (media ≥ 7);

Reprovado (media ≤ 3)

Recuperação (media de 3.1 a 6.9).

1. A empresa “Flor do campo” faz o pagamento de seus funcionários por hora. Faça um algoritmo que calcule e exiba o salário de um funcionário (obter nome, horas trabalhadas e nível do funcionário).

Sabendo que o valor da hora segue a tabela abaixo:

Funcionário Nível Junior R\$12,00 por hora

Funcionário Nível Pleno R\$25,00 por hora

Funcionário Nível Senior R\$50,00 por hora



EXERCÍCIO

3. Dados três valores A , B e C , verifique se eles podem ser os comprimentos dos lados de um triângulo, se forem, verificar o tipo do triângulo. Se os lados não formarem um triângulo, escrever uma mensagem.

O comprimento de cada lado em um triângulo é menor que a soma dos outros dois lados;

Equiláteros: tem os comprimentos dos três lados iguais;

Isósceles: tem os comprimentos de dois lados iguais;

Escaleno: tem os comprimentos dos três lados diferentes



EXERCÍCIO

4. Uma empresa realizará um ajuste salarial variável de acordo com o cargo, conforme a tabela abaixo. Escreva um algoritmo que leia o salário e o cargo de um funcionário e calcule o novo salário. Se o funcionário for de um cargo diferente dos apresentados na tabela, ele deverá então receber 40% de aumento. Mostre o salário antigo, o novo salário e a diferença.

| Código | Cargo | Percentual |
|--------|---------------|------------|
| 1 | Estagiário | 10% |
| 2 | Desenvolvedor | 20% |
| 3 | Analista | 30% |



EXERCÍCIO

5. Uma seguradora concederá um desconto especial para alguns clientes. Faça um algoritmo que leia o valor do seguro de um cliente e calcule o valor de desconto de acordo com a tabela abaixo. Mostre uma mensagem informando o valor do desconto.

Valor o seguro de 0 a 800 nenhum desconto de 801 a 1000 5% de desconto de 1001 a 1500 10% de desconto acima de 1501 15% de desconto.



EXERCÍCIO

6. Somar os valores digitados enquanto o valor for par. No fim do algoritmo apresentar os valores da soma.
7. Entre com 1 valor n e calcule e apresente o valor da tabuada de 1 até esse valor n , conforme o exemplo abaixo:
 $1 \times n = n$
 $2 \times n = 2n$
 $3 \times n = 3n$
...
6. Escrever um algoritmo que leia valores para m e n e que calcule a soma dos valores entre o menor valor de m e n e o maior valor.
7. Escreva um algoritmo que apresente os valores divisíveis por 10 de 100 até 1000.



Conteúdo Aula 5

Estrutura de repetição

1

Repetição com
interrupção no fim

a.

Repetição com
variável de controle

b.



ESTRUTURA DE REPETIÇÃO COM INTERRUPÇÃO NO FIM

algoritmo "RepeticaoInterrupcaoFim"

var

 count : inteiro

inicio

 repita

 Escreval(count)

 count <- count + 1

 ate count > 100

fimalgoritmo



ESTRUTURA DE REPETIÇÃO COM INTERRUPÇÃO NO FIM

Enquanto o resultado da condição for falso irá continuar executando o código entre os comandos **repita...ate.**



ESTRUTURA DE REPETIÇÃO COM VARIÁVEL DE CONTROLE

algoritmo "RepeticaoVariavelControle"

var

 i : inteiro

inicio

 para i de 0 ate 100 passo 1 faca

 Escreval(i)

 fimpara

fimalgoritmo



ESTRUTURA DE REPETIÇÃO COM INTERRUPÇÃO NO FIM

Enquanto o resultado o valor da variável não chegar até o valor informado entre os comandos **de...ate**, irá continuar executando o código entre os comandos **faca...fimpara**.



EXERCÍCIO

1. Escreva um algoritmo que apresente os valores de 100 até 1.
2. Faça um algoritmo que leia um número e mostre a fatorial desse número. ($N * N - 1 * N - 2 * \dots * N - N$)
3. Foi feita uma pesquisa com 5 pessoas de uma cidade e coletados os seguintes dados:
 - a. sexo: M (masculino) e F (feminino)
 - b. cor dos olhos: A (azuis), V (verdes) e C (castanhos)
 - c. idade

Deseja-se saber:

- a maior idade do grupo
- a quantidade de indivíduos do sexo feminino, cuja idade está entre 18 e 35 anos e que tenham olhos azuis.



EXERCÍCIO

4. Escreva um algoritmo que leia 2 valores X e Y e que calcule X na potência Y.
5. Faça um programa que obtenha a idade de 10 pessoa. Calcule e apresente a idade média das 10 pessoas, a maior idade e a menor idade.
6. Obtenha o valor de um produto e apresente o valor das parcelas.

Dado que em 1 parcela: 10% de desconto, de 2 até 5 parcelas 5% de acréscimo e de 6 até 10 parcelas 10% de acréscimo.



EXERCÍCIO

7. Escreva um algoritmo dado o nome e o seu salário bruto de um funcionário considere:

- a. O salário bruto teve um reajuste de 38%.
- b. O funcionário receberá uma gratificação de 20% do salário bruto.
- c. O Salário total é descontado em 15%
- d. Imprimir Nome, idade e cargo.
- e. Imprimir o salário bruto.
- f. Imprimir o salário líquido.



Conteúdo Aula 6

1. Vetor



Conteúdo Aula 6

Vetor

1



VETOR

```
algoritmo "Vetor"  
var  
    nota1, nota2, nota3 : inteiro  
inicio  
    Escreval("Nota 1: ")  
    leia(nota1)  
    Escreval("Nota 2: ")  
    leia(nota2)  
    Escreval("Nota 3: ")  
    leia(nota3)  
    Escreval(nota1)  
    Escreval(nota2)  
    Escreval(nota3)  
finalgoritmo
```



VETOR

```
algoritmo "Vetor"
var
    notas : vetor [0..2] de inteiro
    i : inteiro
inicio
    para i de 0 ate 2 passo 1 faca
        Escreval("Digite a nota: ")
        leia(notas[i])
    fimpara
    para i de 0 ate 2 passo 1 faca
        Escreval(notas[i])
    fimpara
finalgoritmo
```



VETOR

Vetor são estruturas de dados simples que podem nos ajudar muito quando utilizamos variáveis de um mesmo tipo.

Ex: Lista de cidade por estado.

O vetor é um tipo homogêneo, o que significa isso?

Armazenam apenas valores do mesmo tipo.

Vetor deve iniciar sempre no índice 0, porém no visualg aceita que inicie em qualquer posição.



EXERCÍCIO

1. Escrever um algoritmo que possua um vetor inteiro de 5 posições com os dados fornecidos pelo usuário. Após preencher o vetor apresentar a posição e o valor preenchido na posição.
2. Escreva um algoritmo que armazene em um vetor o quadrado dos números ímpares de 1 a 10. Imprima todos os valores armazenados.
3. Escreva um algoritmo que receba 10 números obtidos do usuário e armazene em um vetor a metade de cada número. Imprima todos os valores armazenados.



EXERCÍCIO

4.3) Faça um procedimento que recebe a idade de um nadador por parâmetro e retorna , também por parâmetro, a categoria desse nadador de acordo com a tabela abaixo:

- a. Idade - Categoria
- b. 5 a 7 anos - Infantil A
- c. 8 a 10 anos - Infantil B
- d. 11-13 anos - Juvenil A
- e. 14-17 anos - Juvenil B
- f. Maiores de 18 anos (inclusive) - Adulto



Conteúdo Aula 7

Matriz

1



MATRIZ

```
algoritmo "Matriz"  
var  
    notas : vetor [0..2, 0..1] de inteiro  
    l, c : inteiro  
inicio  
    para l de 0 ate 2 passo 1 faca  
        para c de 0 ate 1 passo 1 faca  
            Escreval("Digite a nota [", l, "][", c, "]")  
            leia(notas[l, c])  
        fimpara  
    fimpara  
    para l de 0 ate 2 passo 1 faca  
        para c de 0 ate 1 passo 1 faca  
            Escreval("Digite a nota [", l, "][", c, "] = ", notas[l, c])  
        fimpara  
    fimpara  
fimalgoritmo
```



MATRIZ

Matriz pode se dizer que é uma lista de vetores.

Ex: Lista agenda de compromissos.

A matriz é um tipo de dado homogêneo.

Matriz deve iniciar sempre no índice 0, porém o visualg aceita que inicie em qualquer posição assim como no vetor.



EXERCÍCIO

1. Escrever um algoritmo que crie uma matriz de 4 x 4 preencha os valores da matriz com informações do usuário e após verifique qual o maior valor e qual o menor valor presente na matriz.
2. Escreva um algoritmo que multiplique os valores de 2 matrizes 2 x 2.
3. Escreva um algoritmo que preencha os valores de uma matriz 5 x 5 e após a matriz estar preenchida deve apresentar os valores da diagonal da matriz. Começando pelo valor na posição 0 x 0.
4. Escrever um algoritmo que preencha os valores de uma matriz 5x5. Após a matriz estar preenchida realizar a soma de todos os elementos da matriz e apresentar o resultado na tela.
5. Crie um algoritmo que armazenará 5 notas de 10 alunos em uma matriz.



EXERCÍCIO

6. Escrever um algoritmo que crie uma matriz de 4 x 4 preencha os valores da matriz com informações do usuário e obtenha o número de uma linha do usuário para realizar a soma dos valores da linha informada.
7. Crie uma matriz onde cada elemento terá o valor da soma dos seus índices:

$$0 \times 0 = 0$$

$$0 \times 1 = 1$$

...



Conteúdo Aula 8

Procedimentos e
funções **1**

Escopo de
variáveis **2**



PROCEDIMENTOS E FUNÇÕES

```
algoritmo "Procedimento"  
var  
    num1, num2, resultado : inteiro  
inicio  
    num1 <- 1  
    num2 <- 2  
    resultado <- num1 + num2  
    Escreval(resultado)  
    num1 <- 4  
    num2 <- 5  
    resultado <- num1 + num2  
    Escreval(resultado)  
finalgoritmo
```



PROCEDIMENTOS E FUNÇÕES

```
algoritmo "Procedimento"  
var  
    num1, num2, resultado : inteiro  
procedimento soma()  
inicio  
    resultado <- num1 + num2  
    Escreval(resultado)  
fimprocedimento  
inicio  
    num1 <- 1  
    num2 <- 2  
    soma()  
    num1 <- 4  
    num2 <- 5  
    soma()  
finalgoritmo
```



PROCEDIMENTOS E FUNÇÕES

```
algoritmo "Funcao"  
var  
    num1, num2: inteiro  
funcao soma() : inteiro  
inicio  
    retorne num1 + num2  
fimfuncao  
inicio  
    num1 <- 1  
    num2 <- 2  
    Escreval(soma())  
    num1 <- 4  
    num2 <- 5  
    Escreval(soma())  
finalgoritmo
```



PROCEDIMENTOS E FUNÇÕES

Procedimento

São trechos de códigos que são executados quando o procedimento é chamado. Procedimento seria uma ordem sem a necessidade de um retorno.

Função

É um tipo de procedimento especial que após a chamada dele, é obrigatório um retorno.

Vantagens

- Reutilização de código.

- Legibilidade do código.

- Divisão do código em parte coerentes.

- Maior facilidade para testes separados.



ESCOPO DE VARIÁVEIS

```
algoritmo "Escopo"  
var  
    num1, num2, resultado: inteiro  
funcao soma() : inteiro  
var  
    resultado : inteiro  
inicio  
    resultado <- num1 + num2  
    retorne resultado  
fimfuncao  
inicio  
    num1 <- 1  
    num2 <- 2  
    resultado <- soma()  
    Escreval(resultado)  
finalgoritmo
```




ESCOPO DE VARIÁVEIS

Variável global

Pode ser utilizada pelo programa principal e por todos os procedimentos e funções pertencentes a ele.

Variável local

Pode ser utilizada apenas pelo procedimento ou função que a criou.



EXERCÍCIO

1. Escreva um algoritmo que irá obter a quantidade de lado e escreverá no console:

3 lados - Triângulo

4 lados - Quadrado

5 lados - Pentágono

Mais do que 5 ou menos do que 3 dados inválidos.

1. Escreva um algoritmo de um calculadora, que irá solicitar ao usuário qual operação deseja realizar. De acordo com a operação desejada será chamado um procedimento diferente que obterá 2 valores, realizará o calculo da operação informado e escreverá o resultado.

2. Escreva uma função que retorna o número maior de 2 números obtidos do usuário.



EXERCÍCIO

4. Crie um menu com as opções:

- 1 - Inserir
- 2 - Exibir
- 3 - Sair

Crie um procedimento para inserir e um para exibir conteúdo do vetor e ao digitar a opção 3 deve parar de executar o algoritmo.

4. Escreva um algoritmo com 2 procedimentos, um para preencher o vetor de 10 posições e outro que realize a soma dos valores desse vetor e apresente o valor na tela.

5. Calcular a soma dos números ímpares de um intervalo informado pelo usuário.

6. Escreva uma função que retornará se o número informado pelo usuário é par ou ímpar.



Conteúdo Aula 9

Utilização de
parâmetros

1

Recursividade

2

Refinamento
sucessivo

3



UTILIZAÇÃO DE PARÂMETROS

```
algoritmo "Parametros"  
var  
    num1, num2, resultado: inteiro  
funcao soma(numero1, numero2 : inteiro) : inteiro  
var  
    resultado : inteiro  
inicio  
    resultado <- numero1 + numero2  
    retorne resultado  
fimfuncao  
inicio  
    num1 <- 1  
    num2 <- 2  
    resultado <- soma(num1, num2)  
    Escreval(resultado)  
finalgoritmo
```



UTILIZAÇÃO DE PARÂMETROS

Parâmetros

São valores que são passados de uma função/procedimento para outro, a fim de tentar evitar problemas com a utilização de variáveis globais.

Esses parâmetros são variáveis, portanto deve ser informado um tipo de dados a eles.



RECURSIVIDADE

```
algoritmo "Fatorial"  
var resultado : inteiro  
funcao fatorial(numero :inteiro) : inteiro  
inicio  
    se numero = 1 entao  
        retorne numero  
    senao  
        retorne numero * fatorial(numero - 1)  
fimse  
fimfuncao  
  
inicio  
    resultado <- fatorial(4)  
    escreva("O fatorial de ", 4," é ", resultado)  
finalgoritmo
```



RECURSIVIDADE

Quando o problema que encontramos possui a mesma solução dos subproblemas encontrados, adotamos a solução como sendo a recursividade.

Na recursividade o processamento para quando o caso base for atingido, por exemplo na fatorial o caso base pode ser o valor ser 1, pois sabes o resultado, fatorial de 1 é 1.

Ex: Fatorial, Potência, Tabuada, utilização de estrutura de árvores em geral.



REFINAMENTO SUCESSIVO

Também conhecida como **dividir para conquista**.

Essa técnica é utilizada dividindo um problema em pequenos problemas e após todos estarem resolvidos, todos são integrados.

Qual o problema encontrado?

Pegar 1 como de água.

Refinando o primeiro nível.

Pegar 1 copo.

Encher ele com água.



REFINAMENTO SUCESSIVO

Refinamos mais um pouco.

Pegar o copo.

Colocar o copo no filtro.

Pressionar o botão para a água sair.

Encher o copo com água.

E assim continua, sempre tentando minimizar cada vez mais os nossos problemas para conseguir resolver todos com mais precisão.



EXERCÍCIO

1. Escreva um algoritmo com uma função que receba por parâmetro um número e retorne se este número é par ou ímpar.
2. Escreva um algoritmo que receba 2 notas por parâmetro, calcule a média das notas e escreva “Parabéns você foi aprovado!”, caso a média seja maior do que 7.
3. Escreva uma função para retornar a soma dos valores de 0 até o valor informado pelo usuário. Utilize função recursiva para realizar esse algoritmo.
4. Utilizando função recursiva, faça o cálculo da potência de um número. Deve ser obtido do usuário o número e a potência.
5. Escreva um algoritmo que da um vetor de 5 posições preenchido, retorne qual o índice que se encontra um elemento solicitado pelo usuário. Caso esse valor não exista retornar -1.



EXERCÍCIO

6. Crie um algoritmo para apresentar a tabuada, dado que a função receberá qual o valor n que será feita a tabuada com base nesse valor:

$$1 * N = 1N$$

$$2 * N = 2N$$

$$3 * N = 3N$$

...

Utilize recursividade para trabalhar com esses valores.