

Lógica e Algoritmo

Prof. David S. Tosta

Agenda

- Revisão
 - Algoritmo
 - Raciocínio lógico
 - Estruturas de decisão
 - Estruturas de repetição

Algoritmo

- Algoritmos é uma maneira de formalizar uma sequência de passos para execução de uma atividade.
- Desta forma um algoritmo precisa:
 - Ter início e fim
 - Ser descrito em termos de ações não ambíguas e bem definidas
 - Que as ações sigam uma sequência ordenada

Algoritmo

- Sequência Lógica:
 - São os diferentes passos ou instruções para solucionar um problema;
- Instruções:
 - São um conjunto de regras ou normas definidas para realização de uma atividade. É uma ação elementar executada pelo computador;

Algoritmo

- Metodologia de Solução
 - 1 - Entender o problema;
 - 2 - Formular um esboço da solução;
 - 3 - Fazer uma primeira aproximação;
 - 4 - Rever os passos originais, detalhando;
 - 5 - Se o algoritmo estiver suficientemente detalhado, testar com um conjunto de dados significativos;
 - 6 - Implementar numa linguagem de programação.

Algoritmo

- Porque o Português Deve Ser Estruturado?
 - Evitar ambiguidades
 - Escritos de forma padronizada para a compreensão do utilizador;
 - Estrutura deve ser semelhante ao código do programa que o representa;
 - Margens representam hierarquicamente suas estruturas;
 - Comentários delimitados.

Algoritmo

- Conceitos Iniciais
 - Identificadores
 - 1. O primeiro caractere deve ser uma letra
 - 2. Os nomes devem ser formados por caracteres pertencentes ao seguinte conjunto :
 $\{a,b,c,...z,A,B,C,...Z,0,1,2,...,9,_ \}$
 - EX: A, BI, BC3D, SOMA, CONTADOR
 - Constantes - É um identificador que armazena um valor fixo e imutável, durante a execução de um algoritmo ou programa. EX: $\pi = 3,1416$

Algoritmo

- Conceitos Iniciais
 - Variáveis - Uma variável (Var) é um identificador que, como sugere o nome, possui o conteúdo variável durante a execução de um algoritmo ou programa.
 - Tipo de dados - uma combinação de valores e de operações que uma variável pode executar

TIPO	DESCRIÇÃO
INTEIRO	Representa valores inteiros. Exemplos: 10, 5, -5, -10
REAL ou NUMERICO	Representa valores reais (com ponto separador da parte decimal). Exemplos: 10, 15.5, -14.67
LITERAL ou CARACTERE	Representa texto (seqüência ou cadeia de caracteres) entre aspas duplas. Exemplo "Esta é uma cadeia de caracteres", "B", "1234"
LOGICO	Representa valores lógicos (VERDADEIRO ou FALSO).

Algoritmo

- Declaração de variáveis
 - Consiste na definição dos nomes e tipos das variáveis que serão utilizadas pelos algoritmos, previamente à sua utilização, incluindo comentário, quando se fizerem necessários. Ex: Inteiro SOMA;
- Palavras Reservadas
 - São palavras que terão uso específico no nosso pseudocódigo e que não deverão ser usadas como identificadores, para não causar confusão na interpretação.

Algoritmo

- Operadores

- Na solução da grande maioria dos problemas é necessário que as variáveis tenham seus valores consultados ou alterados.
- Operadores Aritméticos

Operadores	Exemplo
+ (Adição)	Adicionar duas variáveis : $var1 + var2$
- (Subtração)	Subtrair duas variáveis: $var1 - var2$
- (Menos Unário)	$var = 5$; $var1 = - var$
* (Multiplicação)	Multiplicação de duas variáveis: $var1 * var2$
/ (Divisão)	Divisão de duas variáveis: $var1 / var2$
^ (Exponenciação)	Exponenciação : $var1 ^ 4$
MOD (Resto da Divisão)	Resto : $var1 \text{ } \beta \text{ } 5 \text{ Mod } 4$; $var1 \text{ } \beta \text{ } 1$
TRUNC(A)	A parte inteira de um número fracionário
ARREDONDA(A)	Transforma por arredondamento um número fracionário em um número inteiro
SINAL(A)	Fornece o valor -1, 1 ou 0 conforme o valor de A seja negativo, positivo ou 0

Raciocínio Lógico

- A Lógica - É o ramo da filosofia que cuida das regras do raciocínio.
- Proposição - É uma frase que se pretende ou verdadeira ou falsa, não podendo haver uma terceira opção. Toda proposição é uma frase, mas nem toda frase é uma proposição; uma frase é uma proposição apenas quando possui valor de verdade (possibilidade de ser VERDADEIRA ou FALSA).

Ex: O homem é mortal – Proposição

Abra a porta! - frase IMPERATIVA, portanto não é PROPOSIÇÃO

Qual é o seu nome? - frase INTERROGATIVA, não é PROPOSIÇÃO,

Raciocínio Lógico

- Argumentos - Um argumento é constituído por um conjunto de proposições que pretendem provar/demonstrar uma ideia/tese.
- Um tipo de argumento é o silogismo, que é constituído de três proposições declarativas (ou mais) que se conectam de tal modo que a partir das duas primeiras – as premissas -, é possível deduzir a terceira – a conclusão.

Ex: Mariana disse que estaria na biblioteca ou na lanchonete.

Fui até a biblioteca e Mariana não estava lá.

(Logo,) Mariana está na lanchonete.

Raciocínio Lógico

- Operações Lógicas:
 - São usadas para formar novas proposições a partir de proposições existentes
 - Considerando p e q duas proposições genéricas, pode-se aplicar as seguintes operações lógicas básicas sobre elas

Operação	Símbolo	Significado
Negação	\sim	Não
Conjunção	\wedge	E
Disjunção	\vee	OU

Raciocínio Lógico

- Exemplos de aplicação das operações lógicas :
 - Considere:
 - $p = 7 \text{ é primo} = (V)$
 - $q = 4 \text{ é impar} = (F)$
 - Então:
 - $4 \text{ NÃO é impar} = \sim q$
 - $7 \text{ NÃO é primo} = \sim p$
 - $7 \text{ é primo E } 4 \text{ NÃO é impar} = p \wedge \sim q$
 - $7 \text{ é primo E } 4 \text{ é impar} = p \wedge q$
 - $4 \text{ é impar E } 7 \text{ é primo} = q \wedge p$

Raciocínio Lógico

- Tabela Verdade:

p	q	$\sim p$	$p \wedge q$	$p \vee q$
V	V	F	V	V
V	F	F	F	V
F	V	V	F	V
F	F	V	F	F

- Não (\sim) troca o valor lógico. Se é F passa a ser V e vice-versa
- E (\wedge) só tem valor V quando as duas proposições forem V, basta uma proposição ser F para o resultado ser F
- OU (\vee) só tem valor F quando as duas proposições forem F, basta uma proposição ser V para o resultado ser V

Português Estruturado

- Operadores Relacionais

Operador	Exemplo
< (Menor)	var1 < 5; Verifica se o conteúdo de var1 é menor que 5
> (Maior)	var1 > 5; Verifica se o conteúdo de var1 é maior que 5
<= (Menor ou Igual)	var1 <= 5; Verifica se o conteúdo de var1 é menor ou igual a 5
>= (Maior ou Igual)	var1 >= 5; Verifica se o conteúdo de var1 é maior ou igual a 5
= (Igual)	var1 = 5; Verifica se o conteúdo de var1 é igual a 5
<> (Desigualdade)	var1 <> 5; Verifica se o conteúdo de var1 é desigual a 5

Português Estruturado

- Operadores Lógicos
E, OU e NÃO

p	q	p E q	p OU q
V	V	V	V
V	F	F	V
F	V	F	V
F	F	F	F

Português Estruturado

- Exercício I
- Considerando $p = V$, $q = F$ e $r = V$, resolva as seguintes expressões lógicas
 - $(p \wedge q) \vee \sim r$
 - $(\sim p) \wedge (q \vee \sim r)$
 - $(p \vee q \wedge r) \wedge (\sim p \vee \sim q)$
 - $(p \wedge \sim p) \vee \sim q$

Português Estruturado

- Comando de Entrada de Dados - Leia(variável)
- Comando de Saída de Dados - Escreva (variável)
- Documentação - * ou /* */

“Escreva os comentários no momento em que estiver escrevendo o algoritmo. Um algoritmo não documentado é um dos piores erros que um programador pode cometer e é sinal de amadorismo (mesmo com dez anos de experiência). Como o objetivo de se escrever comentários é facilitar o entendimento do algoritmo, eles devem ser tão bem concebidos quanto o próprio algoritmo.”

Português Estruturado

- Exercício II

- Elaborar um algoritmo que Calcule a área de uma circunferência. O algoritmo deverá solicitar o raio da circunferência.
- Como calcular a área da circunferência: $A = \pi r^2$, considere o valor de $\pi = 3,1416$

Português Estruturado

- Exercício III

- Faça um algoritmo que receba um valor que foi depositado e exiba o valor com rendimento após 4 meses. Considere fixo o juro da poupança em 0,30% a. m

Boas Práticas

- Procure incorporar comentários no algoritmo, pelo menos para descrever o significado das variáveis utilizadas. Comentários em PORTUGOL, podem ocorrer em qualquer parte do algoritmo.
- Escolha nomes de variáveis que sejam significativos, isto é, que traduzam o tipo de informação a ser armazenada na variável.
- Destaque todas as palavras-chave com letras maiúsculas para que ressalte as estruturas de controle do algoritmo.

Boas Práticas

- Procure alinhar os comandos de acordo com o nível a que pertençam, isto é, destaque a estrutura na qual estão contidos.
- É recomendado que todo algoritmo ou procedimento tenha comentários no seu prólogo para explicar o que ele faz e fornecer instruções para seu uso.
- Utilize espaços em branco para melhorar a legibilidade. Espaços em branco, inclusive linhas em branco, são valiosíssimos para melhorar a aparência de um algoritmo.

Boas Práticas

- Um comando por linha é suficiente. A utilização de vários comandos por linha é prejudicial por várias razões.
- Utilize parênteses para aumentar a legibilidade e prevenir-se contra erros.
- Utilize “identação” para mostrar a estrutura lógica do algoritmo. A identificação não deve ser feita de forma caótica, mas segundo certos padrões estabelecidos.

Estruturas Condicionais

- A Estrutura Condicional possibilita a escolha de um grupo de ações e estruturas a serem executadas quando determinadas condições são ou não satisfeitas.
- Estrutura Condicional pode ser **Simples** ou **Composta**.

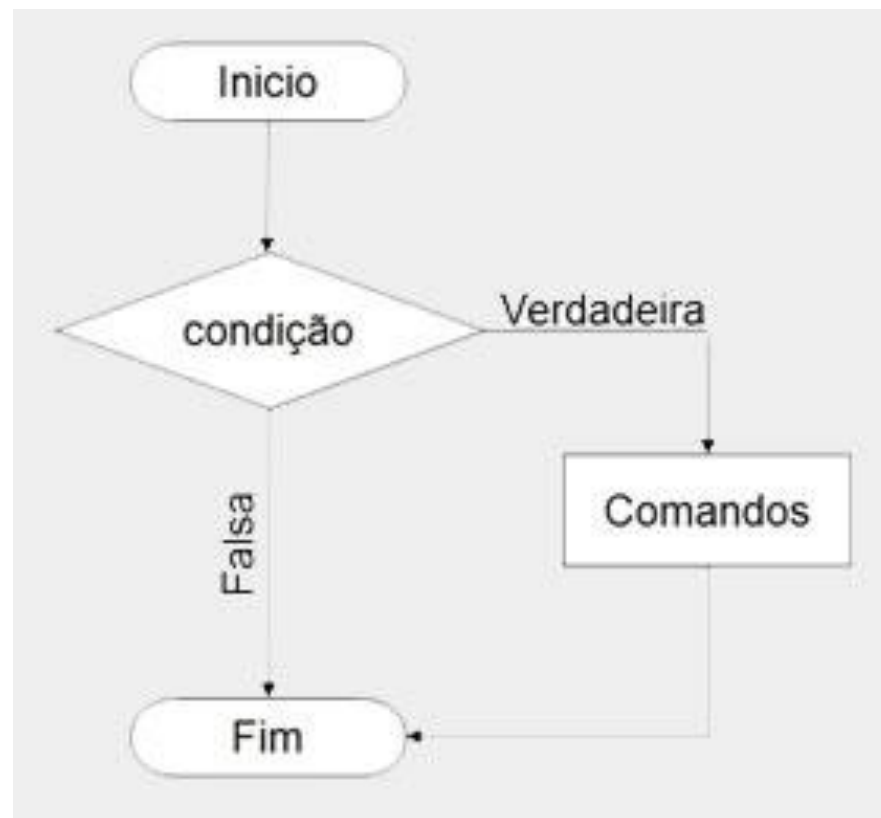
Estruturas Condicionais

- A Estrutura Condicional Simples executa um comando ou vários comandos se a condição for verdadeira. Se a condição for falsa, a estrutura é finalizada sem executar os comandos.
- O comando que define a estrutura é representado pela palavra SE.

```
INICIO
  SE <condição> ENTAO
    INICIO
      comando 1
      comando 2
      ....
      comando N
    FIM
  FIM SE
FIM
```

Estruturas Condicionais

- A Estrutura Condicional Simples - fluxograma



Estruturas Condicionais

- Exercício IV

- Crie um algoritmo que leia 4 números e calcule a média destes três números.

$$MEDIA = \frac{N1 + N2 + N3 + N4}{4}$$

- Considerando apenas a parte inteira do valor obtido da média indique se o número é impar, ou par ou zero.

Estrutura Condicional Simples Aninhada

- É possível termos um comando SE “subordinado” a um outro comando SE.

```

INICIO
  SE <condição> ENTAO
    SE <condição> ENTAO
      INICIO
        comando 1
        comando 2
        ....
        comando N
      FIM
    FIM SE
  FIM SE
FIM
  
```

```

INICIO
  SE <condição> ENTAO
    INICIO
      comando 1
      comando 2
      .....
      SE <condição> ENTAO
        INICIO
          comando 1
          comando 2
          ....
          comando N
        FIM
      FIM SE
    FIM
  FIM SE
FIM
  
```

Português Estruturado

- Exercício V

- Construa um algoritmo que leia três números inteiros e verifique se algum destes números é um numero par, caso algum deles seja par verifique se é divisível por 4.

Estrutura Condicional Composta

- A Estrutura Condicional Composta - Quando se executa um comando (ou sequência de comando) se uma condição é verdadeira, e se executa um outro comando (ou sequência de comandos) se a condição é falsa.

```
INICIO
  SE <condição> ENTAO
    INICIO
      comando 1
      comando 2
      ....
      comando N
    FIM
  SENÃO
    INICIO
      comando 1
      comando 2
      ....
      comando N
    FIM
  FIM SE
FIM
```

Estrutura Condicional Múltiplo

- A Estrutura Condicional Múltiplo - É uma solução elegante quanto se tem várias estruturas de decisão (SE-ENTÃO-SENÃO) aninhadas. Isto é, quando outras verificações são feitas caso a anterior tenha falhado (ou seja, o fluxo do algoritmo entrou no bloco SENÃO). A proposta da estrutura ESCOLHA-CASO é permitir ir direto no bloco de código desejado, dependendo do valor de uma variável de verificação.

```
ESCOLHA <variável de verificação>  
    CASO <valor1> FAÇA  
        <Comandos>  
    CASO <valor2> FAÇA  
        <Comandos>  
    OUTRO CASO  
        <Comandos>  
FIM ESCOLHA
```


Estrutura Condicional Múltipla

- Exercício VI

- Construa um algoritmo que leia um numero de 1 a 6 e mostre os primeiro 6 meses do ano . Considere que:
 - 1 – Janeiro
 - 2 – Fevereiro
 - 3 – Março
 - 4 – Abril
 - 5 – Maio
 - 6 – Junho
- Caso o número seja diferente de (1 a 6) informe que o número não é valido.

Estrutura de Repetição - Enquanto

- A estrutura de repetição (enquanto) é utilizada quando um conjunto de comandos deve ser executado repetidamente, enquanto uma determinada condição (expressão lógica) permanecer verdadeira.
- Dependendo do resultado do teste da condição, o conjunto de comandos poderá não ser executado nem uma vez (se for falsa no primeiro teste), ou será executado várias vezes (enquanto for verdadeira). Chama-se a isso um laço ("loop").
- Da mesma forma que a estrutura de seleção, ela permite o aninhamento de repetições, ou seja, a existência de uma estrutura de repetição dentro de outra. Poderão haver também aninhamentos de seleções dentro de estruturas repetitivas e vice-versa.

Estrutura de Repetição - Enquanto

- Dois cuidados ao criar estruturas de repetição (enquanto):
 - Inicializar a(s) variável(eis) que controla(m) o laço antes do início do laço;
 - Inicializar a(s) variável(eis) que controla(m) o laço dentro do laço (seja por leitura ou por atribuição), pois se isto não for feito cairemos no que chamamos um laço infinito e de lá o nosso programa não sairá.

```
ENQUANTO <condição> FAÇA  
    INICIO  
        <comandos>  
    FIM  
FIM ENQUANTO
```

Estrutura de Repetição - Enquanto

- Exercício VII
 - Construa um algoritmo que leia e imprimir os 10 primeiros nomes e idades.

Estruturas de Repetição - Repita ... Até

- Quando se deseja executar a série de comandos uma vez pelo menos, pode se fazer o teste no final. Inicializar a(s) variável(eis) que controla(m) o laço antes do início do laço;
- Uma vantagem do repita é que não é preciso inicializar a(s) variável(eis) de controle do laço antes de entrar no mesmo.
- Deve-se, contudo, ter o cuidado de modificá-la(s) dentro do laço para que não caiamos em um laço infinito
- Dependendo da resposta, fica repetindo o processo até o teste lógico dar Verdadeiro.

Estruturas de Repetição - Repita ... Até

- Os comandos são executados pelo menos uma vez.
- Quando a condição é encontrada, ela é testada, se for verdadeira o comando seguinte será executado, se for falsa, os comandos são reexecutados até que a condição se torne verdadeira.

```
REPITA  
  INICIO  
    <comandos>  
  FIM  
ATE <condição>
```

Estruturas de Repetição - Repita ... Até

- O comando repita-até é equivalente ao comando enquanto, conforme será mostrado no exemplo abaixo.

```
INICIO
  INTEIRO x;
  x := 2;
  REPITA
    INICIO
      IMPRIMA(x);
      x := x+1;
    FIM
  ATE (x>=10);
  IMPRIMA(x);
FIM
```

```
INICIO
  INTEIRO x;
  x := 2;
  ENQUANTO X < 10 FAÇA
    INICIO
      IMPRIMA(x);
      x := x+1;
    FIM
  FIM ENQUANTO;
  IMPRIMA(x);
FIM
```

- Numa estrutura Enquanto, os comandos são executados 0 ou mais vezes. Numa estrutura Repita, os comandos são executados 1 ou mais vezes.

Estrutura de Repetição – Para Faça

- Repete o bloco de comandos enquanto a variável de controle for menor ou igual ao valor final (vlr_fim);
- A variável de controle recebe um valor inicial (vlr_ini) e é incrementada automaticamente pelo parâmetro de incremento padrão (acrécimo de 1), quando a cláusula PASSO é omitida, ou pelo valor definido pelo usuário através desta cláusula.
- A variável de controle NÃO pode ser modificada no bloco de comandos.

```
PARA <var_controle> = vlr_ini ATE vlr_fim PASSO <inc> FAÇA  
    INICIO  
        <comandos>  
    FIM  
FIM PARA
```


Estrutura de Repetição – Para Faça

- Exercício VIII
 - Construa um algoritmo que calcule o número a somo dos 10 primeiros números pares.