

Lógica – Aula 2

Técnicas de Programação

Criando algoritmos

- Regras
- Métodos
- Identificadores
- Variáveis
- Constantes
- Tipos de dados
- Comandos de atribuição
- Operadores aritméticos, relacionais e lógicos

©2009

Profª Maria Marli Milan Luqueta

Criando algoritmos

- Toda linguagem é composta de sintaxe (instruções) e semântica (dados), onde a sintaxe corresponde à forma e a semântica corresponde ao conteúdo.
- Vocês devem aprender a sintaxe dos comandos, mas a principal preocupação deve ser de “como usar esses comandos”.

©2009

Profª Maria Marli Milan Luqueta

Regras para criação de bons algoritmos

- Use comentários com frequência e sempre atualizá-los{ }.
- Escolha nomes de variáveis (elementos básicos que um programa manipula) significativos, todavia evite nomes muito longos.
- Destaque as palavras-chave das estruturas de controle e comandos com sublinhado.
- Utilize espaços e linhas em branco para melhorar a legibilidade.
- Coloque apenas um comando por linha.
- Use identificação nos comandos de acordo com o nível que estejam.

©2009

Profª Maria Marli Milan Luqueta

Método para desenvolvimento de algoritmos

- Faça uma leitura de todo o problema até o final, a fim de formar a primeira impressão. A seguir, releia o problema e faça anotações sobre os pontos principais.
- Verifique se o problema foi bem entendido.
- Extraia do problema todas as suas saídas.
- Extraia do problema todas as suas entradas.
- Identifique qual é o processamento principal.
- Teste cada passo do algoritmo, com todos os seus caminhos para verificar se o processamento está gerando os resultados esperados.

©2009

Profª Maria Marli Milan Luqueta

Identificadores

- As variáveis, funções e procedimentos que usamos em nossos algoritmos precisam receber um nome (rótulo) – os Identificadores e possuem regras de formação:
 - O primeiro caractere deve ser uma letra.
 - Do segundo caractere em diante são permitidos números e letras.
 - O símbolo de underscore (_) pode ser usado para separar nomes compostos.
 - Não são permitidos espaços, caracteres acentuados e símbolos especiais.
 - Palavras reservadas da linguagem não podem ser usadas (início, fim, for, var, end, begin, para, etc.).
 - Não há distinção entre maiúsculo e minúsculo.

©2009

Profª Maria Marli Milan Luqueta

Constantes

- Constante é um determinado valor fixo, estável que não se modifica ao longo do tempo, durante a execução de um programa. Conforme o seu tipo, a constante é classificada como sendo numérica, lógica e literal.
- **Exemplo de constantes:**

$$Nota1 + Nota2 + Nota3 / 3 \leftarrow \text{Constante}$$

©2009

Profª Maria Marli Milan Luqueta

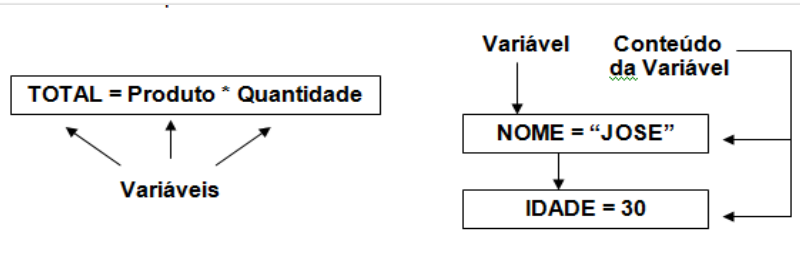
Variáveis

- Variável é a representação simbólica dos elementos de certo conjunto, ou seja, tudo aquilo que é sujeito a variações, que é incerto, inconstante. Cada variável corresponde a uma posição de memória, cujo conteúdo pode se alterado ao longo do tempo durante a execução de um programa. Assim, a variável é o local da memória onde guardamos os dados e o nome da variável é um identificador. Embora uma variável possa assumir diferentes valores, ela só pode armazenar um valor a cada instante.
- Toda variável é identificada pelo programa a partir de seu tipo e pelo seu nome.

©2009

Profª Maria Marli Milan Luqueta

Exemplo de variáveis:



©2009

Profª Maria Marli Milan Luqueta

Variáveis - Declaração

- Toda variável necessita ser declarada, ou seja, reserva-se um local da memória informando que tipo de dados residirão ali. Assim, a sintaxe de declaração de uma variável é:
 - Variavel: tipo de dados ; ou
 - variavel1, variavel2, ..., variaveln: tipo de dados ;

©2009

Profª Maria Marli Milan Luqueta

Tipos de dados

- Ao armazenarmos variáveis na memória do computador, precisamos dizer que tipo de dados elas são, para que seja reservado o espaço adequado, além de ser dado o trabalho correto a elas.
- Estas informações estão caracterizadas por três tipos: dados numéricos (inteiros e reais), dados caracteres ou String e dados lógicos (V ou F).

©2009

Profª Maria Marli Milan Luqueta

Tipo de dados: Numéricos

- Específicos para armazenamento de números, que posteriormente poderão ser utilizados para cálculos.
 - Inteiros: para armazenamento de números inteiros, positivos ou negativos. Ex. -10,0,15,25;
 - Reais: são para o armazenamento de números reais(inteiros, negativos, positivos ou que possuam casas decimais). Ex. -10.5, 1.5,0.3.

©2009

Profª Maria Marli Milan Luqueta

Tipo de dados: Caracteres ou String

- Específicos para armazenamento de conjunto de caracteres que contenham letras, números e símbolos especiais. Uma seqüência de caracteres devem ser indicadas entre aspas. Este tipo de dado também é conhecido como alfa numérico, string, literal ou cadeia.
- Ex. "ABC", "Cidade", "AB2".

©2009

Profª Maria Marli Milan Luqueta

Tipo de dados: Lógico

- Armazenam somente dados lógicos que podem ser Verdadeiro ou Falso, sendo que este tipo de dado poderá representar apenas um dos dois valores. É chamado por alguns autores por tipo booleano.
- Ex. .falso. e .verdadeiro.

©2009

Profª Maria Marli Milan Luqueta

Comentários

- Comentários devem ser inseridos no algoritmo a fim de esclarecer o desenvolvimento do mesmo. Os comentários são inseridos entre { e }.

{ Texto de comentário delimitado por chaves }

©2009

Profª Maria Marli Milan Luqueta

Comandos de atribuição

- ❑ Para atribuirmos um valor ou uma expressão a uma variável, utilizamos o comando de atribuição \leftarrow .
- ❑ Assim, a sintaxe do comando é:
identificador \leftarrow expressão ;
- ❑ Ex.: Nota1 \leftarrow 8,5

Obs.: Em Pascal o símbolo utilizado é :=

©2009

Profª Maria Marli Milan Luqueta

Operadores aritméticos

- ❑ É comum necessitarmos realizar cálculos matemáticos com as informações que estamos manipulando. Para isso, é necessário sabermos qual a representação dos símbolos de operações matemáticas.

©2009

Profª Maria Marli Milan Luqueta

Operadores matemáticos

Operador	Operação	Exemplo
+	Adição	$10 + 15$
-	Subtração	$20 - 10$
*	Multiplicação	$3 * 5$
/	Divisão (onde o resultado será um número real)	$5 / 2 = 2,5$
DIV	Divisão (onde o resultado será um número inteiro)	$10 \text{ div } 2 = 5$ $7 \text{ div } 2 = 3$
MOD	Resto de uma divisão	$7 \text{ mod } 2 = 1$
** ou exp(a, b)	Exponenciação	$5 ** 2$ ou $\text{exp}(5, 2)$

©2009

Profª Maria Marli Milan Luqueta

Operadores Relacionais

- ❑ Além de operações matemáticas, é freqüente nossa necessidade em comparar informações.
- ❑ Por ex.: $A = 5$, $\text{Salario} \geq 5000$, $\text{media} < 5$.

©2009

Profª Maria Marli Milan Luqueta

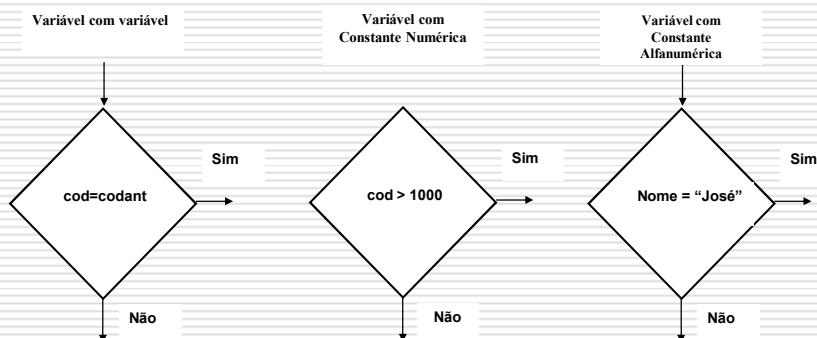
Operadores Relacionais

Operador	Relação
=	Igualdade
≠ ou <>	Diferente
>	Maior que
≥ ou >=	Maior ou igual que
<	Menor que
≤ ou <=	Menor ou igual que

©2009

Profª Maria Marli Milan Luqueta

Símbolo utilizado para comparação entre expressões:



©2009

Profª Maria Marli Milan Luqueta

Operadores Lógicos

- É freqüente precisarmos analisar expressões lógicas, aquelas que só possuem dois valores possíveis: Verdadeiro ou Falso.
- Os operadores lógicos servem para combinar resultados de expressões, retornando se o resultado final é verdadeiro ou falso.

©2009

Profª Maria Marli Milan Luqueta

Operadores Lógicos

Operador	Relação	
E (And)	E lógico	Uma expressão AND é verdadeira se todas as condições forem verdadeiras
Ou (Or)	Ou lógico	Uma expressão OR é verdadeira se pelo menos uma condição for verdadeira
Não (Not)	Negação lógica	Uma expressão NOT inverte o valor da expressão ou condição, se verdadeira inverte para falsa e vice-versa.
Ou-X (Xor)	Ou 'Exclusivo'	Uma expressão Ou-X é verdadeira se dentre as condição somente uma seja verdadeira

©2009

Profª Maria Marli Milan Luqueta

Operadores Lógicos

P	Q	P e Q	P ou Q	P ou-X Q	não P
F	F	F	F	F	V
F	V	F	V	V	--
V	F	F	V	V	F
V	V	V	V	F	--

©2009

Profª Maria Marli Milan Luqueta

Operadores Lógicos

- *Exemplos:* Suponha que temos três variáveis $A = 5$, $B = 8$ e $C = 1$. Os resultados das expressões seriam:

Expressões			Resultado
$A = B$	AND	$B > C$	Falso
$A <> B$	OR	$B < C$	Verdadeiro
$A > B$	NOT		Verdadeiro
$A < B$	AND	$B > C$	Verdadeiro
$A \geq B$	OR	$B = C$	Falso
$A \leq B$	NOT		Falso

©2009

Profª Maria Marli Milan Luqueta

Prioridade na Avaliação de Expressões (hierarquia)

1. Parênteses e funções (resolvidos da esquerda para a direita)
2. Multiplicação (*), Divisão (/ e div) e Resto (Mod) (resolvidos da esquerda para a direita)
3. Soma e subtração
4. Operadores relacionais: >, <, ≥, ≤, =, ≠
5. Operador Lógico Não
6. Operador Lógico E
7. Operador Lógico Ou

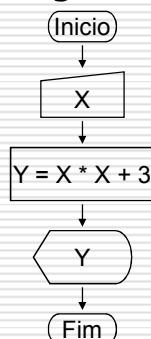
©2009

Profª Maria Marli Milan Luqueta

Exemplo de algoritmo

□ Algoritmo para equação: $y = x^2 + 3 - y$

Fluxograma



Pseudocódigo

```

Programa Exemp01;
Var X,Y: Inteiro;
Inicio
  Leia (X);
  Y=X* X+3;
  Escreva (Y)
Fim
  
```

©2009

Profª Maria Marli Milan Luqueta

Exercícios

- ❑ Criar um algoritmo representado por fluxograma e pseudocódigo para resolver as seguintes equações

b) $y = B + 40$

c) $y = 2x^2 + 3 + 3x + 2$

d) $y = 3 + x + 2/3$

©2009

Profª Maria Marli Milan Luqueta

Exercícios

- ❑ Faça um algoritmo representado por fluxograma e pseudocódigo para "calcular o estoque médio de uma peça", sendo que $ESTOQUE_MÉDIO = (QUANT_MÍN + QUANT_MÁX) / 2$. Faça também o teste de mesa.

©2009

Profª Maria Marli Milan Luqueta

Exercícios

- Desenvolva um digrama de bloco que:
 - Leia 4 (quatro) números;
 - calcule o quadrado para cada um;
 - somem todos;
 - mostre o resultado.
- A partir do exercício anterior, identifique:
 - os dados de entrada;
 - o processamento utilizado;
 - os dados de saída.