

Algoritmos

Prof. Jonatas Bastos

Email: jonatasfbastos@gmail.com

Site: <http://jonatasfbastos.wordpress.com/>

Exemplo de Algoritmo

{ declaração de variáveis }

real: p1, p2, p3, p4, Media;

inicio

{ comandos de entrada de dados }

leia (p1);

leia (p2);

leia (p3);

leia (p4);

{ processamento - Calculo da media }

Media \leftarrow (p1 + p2 + p3 + p4)/4;

{ saída de dados }

escreva (' Media final = ', Media);

fim.

- Um aspecto fundamental na construção de algoritmos computacionais são as estruturas de dados, que **representam** as **informações** do problema a ser resolvido.
- Estruturas de dados são organizadas em tipos distintos de informação
 - **Tipos primitivos de dados**
 - **Tipos estruturados**

Tipos Primitivos de Dados

- Tipos primitivos são os grupos básicos de informações que o computador manipula.
 - **Inteiro**
 - Ex: 1 -4 100 0 -905
 - **Real**
 - Ex: 1,3 816,97 3,0 -0,0055
 - **Caracter**
 - Ex: 'CASA' 'livro' '18' 'R\$ 55,36' 'A'
 - **Lógico**
 - Falso, Verdadeiro (ou False, True)

Exercício Rápido

- Identifique para cada dado abaixo se ele é do tipo Inteiro, Real, Lógico ou Caractere;
 1. Salário do funcionário de uma empresa;
 2. Quantidade de vendedores de uma loja;
 3. Número de latas de refrigerante em uma prateleira;
 4. Nome de um assinante de linha telefônica;
 5. Preço de uma litro de leite;
 6. Estado da iluminação de uma lâmpada em perfeitas condições;

- Outros conceitos relacionados a estruturas de dados
 - **Variável** e **Constante**
- Variáveis representam **informações** cujos **valores** são **modificados** ao **longo da execução** do **algoritmo**.
- Podemos definir variáveis também como um **local** onde um **determinado valor** (de um dos tipos definidos) é **armazenado**.
- Assim, a variável é composta de dois elementos básicos:
 - **Conteúdo**: valor atual da variável
 - **Identificador**: nome dado à variável para possibilitar sua manipulação e está armazenada em um endereço.

- **Representam** valores **constantes**, ou seja, que **não variam** no **decorrer** do **algoritmo**.
- Ex: Um algoritmos para calcular a área de uma circunferência;
 - Teríamos que usar a formula πr^2 , na qual π tem uma valor constante 3,1416..., independente de qual seja a circunferência;
 - Já o valor de r , que representa o valor do raio, é variável, a cada execução do algoritmo.

Declaração de Variáveis

- ❑ As **informações** são guardadas na **memória**;
- ❑ Uma memória pode ser visto como uma **armário** repleto de **gavetas** para **armazenar objetos**;
- ❑ Os **objetos** que podem ser **substituídos**, são os dados das **gavetas** as variáveis;
- ❑ Analogia: Visto que **memória** (armário) existem inúmeras **variáveis** (gavetas), precisamos diferencia-las, o que é feito por meio de **identificadores** (etiquetas ou rótulos); Variável

Endereço
Identificador
Conteúdo

X
2
A
3 56
SOMA
14.283
NUM
8

Declaração de Variáveis

- ❑ Usem o bom senso na declaração de variáveis;
- ❑ Declaração

var

nome, nomeFuncionario, endereco: **caractere**;

idade, soma: **inteiro**;

resposta: **logica**;

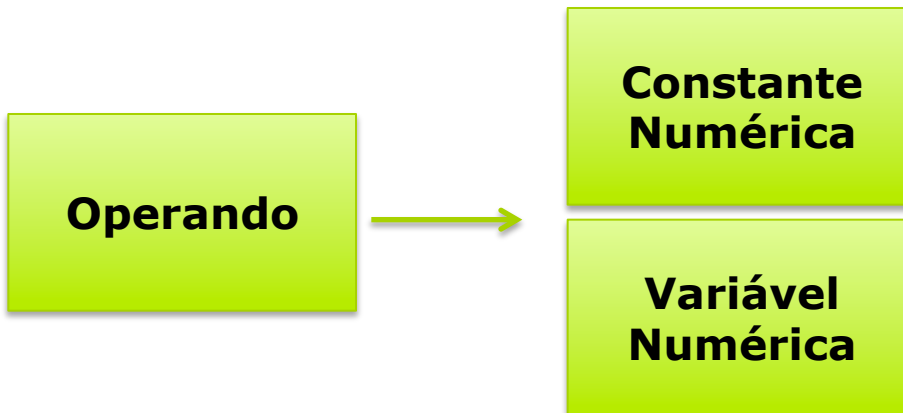
- ❑ O **identificador** idade é o **nome** do local da memória que só pode conter valores do **tipo inteiro**;
- ❑ Não podemos permitir que mais de uma **variável** (gaveta) possua o **mesmo identificador**;
- ❑ Cada **variável** (gaveta) pode guardar no **máximo** um único **objeto por vez**;
- ❑ Neste momento, as **variáveis** ainda não **contém** nenhum valor associado a elas;

Exercícios Rápidos

1. Quais identificadores são válidos:
a) (X) b) u2 c) AH! d) “aluno” e) #55
2. Supondo que as variáveis nb, na, nMat, sx, sejam utilizadas para armazenar a nota do aluno, nome do aluno, número de matrícula e sexo, declare-as corretamente, associando o tipo primitivo adequado do dado que será armazenado;

Expressões Aritméticas

- Expressão aritmética é aquela cujos os **operadores** são **aritméticos** e cujos **operandos** são **constantes** ou **variáveis** do **tipo numérico** (**inteiro** ou **real**);



Operadores Aritméticos

Operador	Função	Exemplos
+	Adição	$2 + 3$, $X + Y$
-	Subtração	$4 - 2$, $N - M$
*	Multiplicação	$3 * 4$, $A * B$
/	Divisão	$10/2$, $X1/X2$

Operador	Função	Significado	Exemplos
pot(x,y)	Potenciação	x elevado a y	pot(2,3)
rad(x)	Radiciação	Raiz quadrada de x	rad(9)

Operadores Aritméticos

Operador	Função	Exemplos
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

- Na resolução de expressões aritméticas, as operações guardam uma hierarquia entre si;

Prioridade	Operadores
1 ^a	parênteses mais internos
2 ^a	pot rad
3 ^a	* / div mod
4 ^a	+ -

- Em caso de **empate** (operadores da mesma prioridade) devemos resolver da **esquerda** para a **direita**;
- Ex:
 - $5 + 9 + 7 + 8/4; \Rightarrow 5 + 9 + 7 + 2 \Rightarrow 23$
 - $1 - 4 * 3/6 - \text{pot}(3,2)$

Operadores Relacionais

- Utilizamos para realizar **comparação** entre dois **valores do mesmo tipo primitivo**;

Operador	Função	Exemplos
=	Igual a	$3 = 3$, $X = Y$
>	Maior que	$5 > 4$, $X > Y$
<	Menor que	$3 < 6$, $X < Y$
>=	Maior ou igual a	$5 >= 3$, $X >= Y$
<=	Menor ou igual a	$3 <= 5$, $X <= Y$
<>	Diferente de	$8 <> 9$, $X <> Y$

- Em caso de **empate** (operadores da mesma prioridade) **devemos** resolver da **esquerda** para a **direita**

- Ex $2 * 4 = 24 / 3$

$$8 = 8$$

V

Operadores Lógicos

- Utilizamos para realizar **comparação** entre **dois valores do mesmo tipo primitivo**;

Operador	Função
não	negação
e	conjunção
ou	disjunção

- Tabela Verdade**

Operação de negação	
A	não A
F	V
V	F

Tabela Verdade

Operação de conjunção

A	B	A e B
F	F	F
F	V	F
V	F	F
V	V	V

Operação de disjunção não-exclusiva

A	B	A ou B
F	F	F
F	V	V
V	F	V
V	V	V

Comando de Atribuição

- ❑ Um comando de atribuição **permite-nos** fornecer um **valor a** uma **variável** (guardar um objeto em uma gaveta) ;
- ❑ O tipo de dado deve ser igual ao da variável;
- ❑ **Ex:**

a, b : logico;

x : inteiro;

a <- b;

x <- 8 + 13 div 5;

b <- 5 = 3;

x <- 2;

Comando de Entrada e Saída

- ❑ Algoritmos **precisam** ser **alimentados** com **dados**;
- ❑ Entrada de dados:
 - O comando para a entrada de dados denomina-se “leia”
 - A finalidade é **atribuir** o **dado** a ser **fornecido** a **variável identificada**;
- ❑ Ex:

leia (x);

leia (a, nota);

Comando de Saída

- ❑ Para que o algoritmo possa **mostrar** os **dados** que **calculou**, como **resposta** ao **problema** que resolveu adotamos o comando de saída;
- ❑ Comando de Saída é o “**escreva**”;
- ❑ **Ex:**
 - escreva (y);**
 - escreva (b, media, soma/4);**
 - escreva (“bom dia”);**
 - escreva (“Você pesa”, peso, “quilos”);**

Impressões Complementares

- ❑ A finalidade principal do comando de saída é exibir resultados;
- ❑ Podemos usar o comando de saída para imprimir tanto variáveis, quanto mensagens relativas ao processamento;
- ❑ Ainda podemos utilizá-lo para mensagens informativas para o usuário, ajudando que ele entenda melhor o que a máquina esta pedindo;
- ❑ Chamamos essas informações de complementares, por não afetarem a lógica do algoritmo;

Impressões Complementares

Ex:

//Algoritmo para calcular média

var

num1, num2, soma, media: real;

inicio

escreva(“Informe o primeiro número:”);

leia(num1);

escreva(“Informe o segundo número:”);

leia(num2);

soma <- num1 + num2;

media <- soma / 2;

escreva(“A média calculada foi”, media);

fim.

Sintaxe Geral do Algoritmo

var

Declaração de Variáveis

//Declaração de variáveis

Comentário

<variavel1>:<tipo1>;

<variavel2>:<tipo2>;

....

//Corpo do Algoritmos

Parte Principal do Algoritmo

inicio

<comando1>;

<comando2>;

Separador de Comandos

...

fim.