

LÓGICA DE PROGRAMAÇÃO

Algoritmos Computacionais

Sérgio Carlos Portari Júnior

portari.uemgfrutal@gmail.com

Tópicos abordados

● Algoritmos Computacionais

● Estrutura de Dados

- Tipos Primitivos
- Constantes
- Variáveis
 - Identificadores
 - Declaração de variáveis
- Comandos de atribuição
- Operadores matemáticos e relacionais e lógicos

● Comandos de Entrada e Saída

● Estrutura de um algoritmo

Algoritmos Computacionais

- Vimos que o algoritmo é uma seqüência de instruções, onde cada instrução representa uma **AÇÃO** que deve ser entendida e realizada. Surge então uma Questão:
 - **Como saber se já temos detalhes suficientes em um algoritmo para que a **AÇÃO** possa ser entendida e realizada?**
- Vai depender do agente que irá executar o Algoritmo. No caso de **algoritmos computacionais**, sabemos que o computador possui um conjunto limitado de instruções e o algoritmo deve ser expresso nos termos destas instruções.

Algoritmos Computacionais

● O computador utiliza dois conceitos básicos para construir e interpretar algoritmos:

- Estruturas de Dados para manipulação das informações.
- Estruturas de Controle para manipulação das ações.

Estrutura de Dados

- Um aspecto fundamental na construção de algoritmos computacionais são as **estruturas de dados**, que representam as informações do problema a ser resolvido.
- Tais estruturas estão organizadas em tipos distintos de informações.
- Dentro do escopo das estruturas de dados, definiremos termos como **CONSTANTE, VARIÁVEL, IDENTIFICADOR, ETC.**

Tipos Primitivos de Dados

● São os grupos de informações que o computador manipula. Podem ser:

- Numéricos

- a) Inteiros Ex: 1 -4 100 0 -905 ...
- b) Reais Ex: 1,3 816,97 3,0 -0,0055 ...

- Não-numéricos

- a) alfanuméricos Ex: "CASA" "livro" "18" "R\$ 55,36" 'a' 'S'
- b) lógicos ou booleanos Ex: Falso, Verdadeiro (ou False, True)

Obs: O delimitador para alfanuméricos pode ser: " " ou ' ' dependendo da ocasião

Constantes

- Representam valores constantes, ou seja, que não variam no decorrer do algoritmo.

● Ex:

- 148
- “Opção:”
- -8.69
- “Tecle algo para continuar”
- Falso
- ‘A’

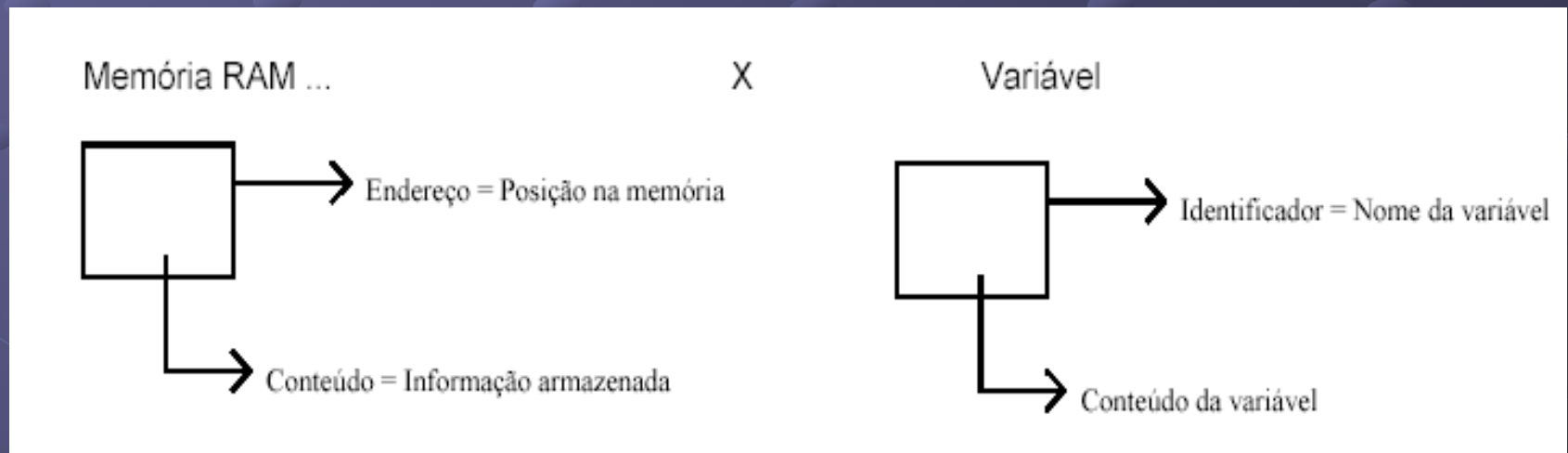
Variáveis

- Representam informações cujos valores são modificados ao longo do tempo (Ex. Genérico: a idade de um indivíduo). 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

Variáveis

- O conceito de **variável**, na verdade, corresponde a “posições de memória RAM”, onde serão armazenados os dados manipulados pelo programa quando este for implementado.



Identificador

- Nome de um local onde se pode colocar qualquer valor do conjunto de valores possíveis de um tipo básico associado. Usado para manipular todos os dados variáveis do algoritmo. Pode também ser usado para rotular valores constantes (ex: uso do identificador PI para representar o valor constante 3,14).

Regras para Definição de Identificadores em Pseudolinguagem (Portugol):

- devem começar por um caractere alfabético (uma letra);
- pode ser seguido de letras e/ou dígitos;
- é permitido o uso do caractere especial “_” situado entre letras e/ou dígitos (números)
- Não pode ser uma palavra reservada pela linguagem (ex: true, false)

Identificador

- Ex. de identificadores válidos:
 - Operador, Nome, X, y, ENDERECO, Aluno_01, L36
- Ex. de identificadores inválidos:
 - 1Nome, 0983, _teste, 3, True, ?teste, *casa
- Obs: O identificador deve representar de forma significativa o conteúdo desejado, ou seja, deve ser um **MNEMÔNICO**.

Declaração de variáveis

- É a criação (ou definição) de locais na memória rotulados com o identificador da variável (ou constante) que será utilizada no algoritmo para a manipulação de um determinado tipo de informação.
- A sintaxe geral em Portugol para declaração de variáveis é a seguinte:

Lista de Variáveis : Tipo ;

- O Tipo pode ser inteiro, real, caracter (alfanumérico) ou booleano.

Declaração de variáveis

● Ex:

- NUM, X : inteiro
- parcela : real
- resposta : booleano
- PI : real *// definição de uma constante real*
- NOME, ENDERECO : caracter

- Neste momento, as variáveis ainda não contém nenhum valor associado a elas.
- A declaração de variáveis deve ser feita antes da definição das ações do algoritmo.
- Os comentários representam qualquer texto que explique uma ação ou um dado usado no algoritmo. Eles podem ser colocados em qualquer ponto do algoritmo, utilizando // ou /* ... Comentário ... */ como delimitadores.

Comandos de Atribuição

- Serve para atribuir (ou associar) um valor a uma variável ou constante.
- Sintaxe Geral em Pseudolinguagem:
Identificador := expressão
- onde *expressão* pode ser um(a): Constante, Variável, Expressão matemática, Função matemática, Expressão booleana, etc.

Comandos de atribuição

- Ex: NOME:="Fulano de Tal"
- PI:=3.14
- Erro:=Verdadeiro
- Media:=(P1 + P2)/2

Operadores aritméticos

- São os operadores para processamento de operações matemáticas no algoritmo:
- * multiplicação
- / divisão
- + - soma e subtração
- div divisão truncada (ou inteira) (ex: $x \text{ div } y$)
- mod resto da divisão inteira

Operadores relacionais

● Operadores Relacionais:

● < Menor

● <= Menor ou igual

● > Maior

● >= Maior ou igual

● = Igual

● < > Diferente

Operadores Lógicos

● not Não

● and E

● or OU

Comandos de Entrada e Saída

- Representam as ações básicas de algoritmos para recebimento e apresentação de dados, respectivamente.
- Entrada de Dados: informações que são fornecidas ao programa pelo “usuário” durante a resolução do problema.
- Representação em Portugol:

`leia (identificador, identificador, ...)`

- Ex: `leia (altura, idade, sexo)`

Comandos de Entrada e Saída

- Saída de Dados: informações que são mostradas ao usuário como resposta ao problema.
- Representação em Pseudolinguagem:

Escreva (expressão, expressão, ...)

Ex: escreva (“Sua idade é : ”, idade)
escreva (“Media final = ”, $(P1 + P2)/2$)

Estrutura de um algoritmo

algoritmo “Nome_do_algoritmo”

//função: “função do algoritmo (objetivo)”

//autor: “autor do algoritmo”

//data: “data da última alteração do algoritmo”

var

declara-se todas as variáveis que serão utilizadas neste algoritmo

início

desenvolve o algoritmo nesta seção, começando pela entrada de dados, processamentos e terminando com a saída dos dados

fimalgoritmo

Estrutura de um algoritmo

- Exemplo: Algoritmo em pseudolingagem para o problema do cálculo da média final dos alunos da 3ª Série.

algoritmo “media_final”

//função: “Calcular a média final de um aluno com 4 provas

//autor: “Sérgio Carlos Portari Júnior

//data: “09/03/07”

var

P1, P2, P3, P4, Media : real

inicio

//comandos de entrada de dados

escreva (“Digite as notas das quatro provas: ”)

leia (p1)

leia (p2)

leia (p3)

leia (p4)

//processamento- Calculo da media

media:=(P1 + P2 + P3 + P4)/4

//saída de dados

escreva (“Média final: ”, media)

fimalgoritmo

Exercícios

- 1 – Utilizando tipos primitivos de dados, crie declarações de variáveis que armazenem as seguintes informações: o nome de uma figura geométrica, a quantidade de lados, a área, o perímetro e se a figura é regular ou não.
- 2 - Faça 02 exemplos para cada um dos conceitos abaixo:
 - a) entrada de dados
 - b) saída de dados
 - c) declaração de variáveis
 - d) inicialização de variáveis (atribuição de números)
 - e) atribuição do resultado de uma expressão aritmética a uma variável (duas operações quaisquer)

Exercícios

3 - Realizarei uma viagem de vários dias em meu automóvel, e gostaria de saber a quilometragem média por litro de gasolina. Para isto, anotarei a quilometragem no velocímetro ao sair de viagem, e depois ao chegar; também vou somar toda a gasolina que comprar para o carro. Você poderia fazer um programa que me desse, com estes dados, quantos km fiz, em média, por litro de gasolina?