



Programação de Computadores I

Arrays (Vetores e Matrizes)

PROFESSORA CINTIA CAETANO

Introdução

- ▶ Imagine que nós precisemos declarar 100 variáveis do tipo inteiro.
- ▶ E se além dessas 100 variáveis, precisarmos também 1000 do tipo Literal.
- ▶ Isso poderia ser feito da seguinte forma:
Var
 i1,i2,i3,i4, ... ,i100: Inteiro
 c1,c2,c3, ..., c1000: Literal

Arrays unidimensionais

- ▶ Estrutura simples.
- ▶ Tipo estruturado de dados (VETOR).
- ▶ Formada de elementos de mesmo tipo.
- ▶ Todos os elementos da estrutura são igualmente acessíveis.

Arrays unidimensionais

► Exemplos

Notas

5.0	6.8	9.7	3.2	...	6.9
1	2	3	4	...	n

Nomes

Cláudia	Carla	João	Ana	...	Roberto
1	2	3	4	...	n

Arrays unidimensionais

► Sintaxe

```
<identificador> : vetor [<tamanho>] de < tipo >
```

```
Tamanho [VI..VF]=> Vi= Valor inicial do índice e VF valor Final do índice.
```

Arrays unidimensionais

- ▶ Exemplo

idade: vetor [1..5] de inteiro

nomes: vetor [1..5] de literal

- ▶ A declaração acima corresponde à declaração de 10 variáveis: nomes[1], nomes[2], nomes[3], nomes[4], nomes[5], idades[1], idades[2], idades[3], idades[4] e idades[5].

Arrays unidimensionais

► Atribuição de valores

```
< identificador>[<posição>] <- <valor>
```

► Exemplos:

```
1.  nomes[1] <- "João da Silva"
2.  idades[1] <- 35
3.  nomes[3] <- "Maria Aparecida"
4.  idades[3] <- idades[1]
5.  i <- 5
6.  idades[i] <- 45
```

Cuidado com vetores

- ▶ Caso o seu limite seja ultrapassado, o programa pode travar e junto com ele o sistema.
- ▶ O programa só acusará erro se você tentar atribuir a um vetor um dado qualquer com índice fora da faixa escrita no programa. Mas isso não acontecerá se a origem do índice for alguma expressão.
- ▶ Os arrays são estáticos, ou seja, o comprimento e as dimensões alocadas são fixas em tempo de compilação e não podem ser redefinidas durante a execução do programa.

Exemplo 1

algoritmo "InserindoVetor"

Var

casas: vetor[1..5] de inteiro

i: inteiro

inicio

para i de 1 ate 5 faca

Escreva("Entre o valor: ")

leia(casas[i])

fimpara

fimalgoritmo

Exemplo 2

Algoritmo "ImprimindoVetor"

Var

casas: vetor[1..5] de inteiro

i: inteiro

inicio

para i de 1 ate 5 faca

Escreva("Entre o valor: ")

leia(casas[i])

fimpara

Escreval("Imprimindo Vetor...")

para i de 1 ate 5 faca

Escreva(casas[i], " ")

fimpara

fimalgoritmo

Exemplo 3

algoritmo "Vetor3"

Var

nomes: vetor[1..3] de Literal

notas: vetor [1..3] de real

soma, media: real

i: inteiro

inicio

soma <- 0

para i de 1 ate 3 faca

 Escreva("Entre com o nome do", i, "º aluno: ")

 Leia(nomes[i])

 Escreva("Entre com sua nota: ")

 Leia(notas[i])

 soma <- soma + notas[i]; //acumulador de notas

fimpara

media <- soma/3 //media geral da turma

Escreva("Média geral da turma: ", media:2:2)

Fimalgoritmo

Exemplo 3

algoritmo "VetorInverso"

Var

 a: inteiro

 num: vetor[1..5] de real

Inicio

 Escreva("Digite os números: ")

 Para a de 1 ate 5 faca

 Leia(num[a])

 fimpara

 Para a de 1 ate 5 faca

 Escreva(num[a])

 fimpara

 Escreval("Imprimindo vetor inverso...")

 Para a de 5 ate 1 passo -1 faca

 Escreval("Indice ", a, " => ", num[a])

 fimpara

fimalgoritmo

Exemplo 4

Algoritmo "vetores"

VAR

NUMERO: **VETOR** [1..6] **DE** **REAL**

I, conta_neg: **INTEIRO**

inicio

conta_neg <- 0

para i de 1 ate 6 faca

 leia (NUMERO[i])

se NUMERO[i] < 0 **entao**

 conta_neg <- conta_neg + 1

fimse

fimpara

para i de 1 ate 6 faca

 escreval (NUMERO[i])

fimpara

escreva ("Total de números negativos: ", conta_neg)

fimalgoritmo

Exercício em Sala

- ▶ Faça um algoritmo que copie o conteúdo de um vetor em um segundo vetor.
- ▶ Faça um algoritmo que some o conteúdo de dois vetores e armazene o resultado em um terceiro vetor
- ▶ Faça um algoritmo que faça a união de dois vetores de mesmo tamanho e mesmo tipo em um terceiro vetor com o dobro do tamanho.

Array Bidimensionais

- ▶ São arrays de duas dimensões.
- ▶ Também chamados de **Matrizes**.

	1	2	3	4	5
1					
2					
3					

Array Multidimensionais

► Sintaxe

```
<identificador> : vetor [<tamanho1>,<tamanho2>] de < tipo >
```

```
Tamanho [VI..VF]=> Vi= Valor inicial do índice e VF valor Final do índice.
```

► Exemplo

PESSOAS:VETOR [1..2,1..3] DE CARACTERE

Array Multidimensionais

- ▶ **Atribuição**

- ▶ `<identificador>[<posição 1>,<posição 2>] <- <valor>`

- ▶ **Exemplo**

- ▶ `PESSOAS[1,1]<- “José”`

- ▶ `PESSOAS[1,2]<- “Ana”`

Exemplo 5

algoritmo "Matriz I"

Var

i,j: inteiro

mat: vetor[1..2,1..2] de inteiro

Inicio

para i de 1 ate 2 faca {linhas}

 para j de 1 ate 2 faca {colunas}

 Escreva("Entre com o valor da linha ", i , " - coluna ", j, ": ")

 Leia(mat[i,j])

 fimpara

fimpara

fimalgoritmo

Exemplo 6

► Imprimindo a Matriz

```
para i de 1 ate 2 faca {linhas}
  para j de 1 ate 2 faca {colunas}
    Escreva(mat[i,j], " ")
  fimpara
  Escreval(" ")
fimpara
```

Exemplo 7

- ▶ Algoritmo que lê uma matriz de 3 X 3.
 - ▶ Calcula a soma da linha 3
 - ▶ Calcula a soma da coluna 2
 - ▶ Calcula a soma da diagonal primária
 - ▶ Calcula a soma de todos os elementos da matriz

Exemplo 7

Algoritmo "Matriz I"

Var

somacoluna, somalinha, somadiagonal, somamatriz, i, j: inteiro

mat: vetor[1..3, 1..3] de inteiro

Inicio

somacoluna <- 0

somalinha <- 0

somadiagonal <- 0

somamatriz <- 0

para i de 1 ate 3 faca {linhas}

 para j de 1 ate 3 faca {colunas}

 Escreva("Entre com o valor : ")

 Leia(mat[i,j])

Exemplo 7

```
somamatriz <- somamatriz + mat[i,j]
se (i=3) entao
    somalinha <- somalinha + mat[i,j]
fimse
se (j=2) entao
    somacoluna <- somacoluna + mat[i,j]
fimse
se (i=j) entao
    somadiagonal <- somadiagonal + mat[i,j]
fimse
fimpara
fimpara
```

Exemplo 7

```
para i de 1 ate 3 faca {linhas}
    para j de 1 ate 3 faca {colunas}
        Escreva(mat[i,j], " ")
    fimpara
    Escreval(" ")
fimpara
Escreval("Soma da coluna 2: ", somacoluna)
Escreval("Soma da linha 3: ", somalinha)
Escreval("Soma da diagonal principal: ", somadiagonal)
Escreval("Soma da Matriz: ", somamatriz)
finalgoritmo
```

Exercício em Sala

- ▶ Faça um algoritmo que copie o conteúdo de uma matriz de 3×3 em uma segunda matriz.
- ▶ Faça um algoritmo que some o conteúdo de duas matrizes de 3×2 e armazene o resultado em uma terceira matriz.
- ▶ Faça um algoritmo que some apenas os números pares de uma matriz de 4×2 .