



INSTITUTO FEDERAL
Rio Grande do Sul

Estruturas de Dados Homogêneas

Profa. Fabrícia Py Tortelli Noronha



Estruturas de dados homogêneas

Possibilitam o armazenamento de mais de um valor em uma única variável na memória do computador.

São chamadas **homogêneas** porque os valores que serão armazenados são de um **mesmo tipo de dado**.

As estruturas homogêneas são divididas em:

Unidimensionais  **Vetores**

Multidimensionais  **Matrizes**



VETORES

- ✓ Também chamados de Arrays
- ✓ Estrutura de dados unidimensional
- ✓ Composto por um conjunto de variáveis homogêneas (todas do mesmo tipo), com o mesmo identificador (nome), agrupadas e indexadas

SINTAXE

<nome>: vetor [<n>..<m>**] de <tipo_dado>**

Onde: <nome> = nome do vetor

<n> = índice do primeiro elemento

<m> = índice do último elemento

<tipo_dado> = tipo de dados dos elementos



Exemplo de declaração de Vetor

Variáveis

v1, v2 : vetor [1..5] de inteiro

V1	1	2	3	4	5

V2	1	2	3	4	5

nota: vetor [1..50] de real

nota	1	2	3	4	5	6	...	48	49	50

EXEMPLO

nota: vetor [1.. 50] de real

nota[1] <- 10,0

nota[2] <- 8,5

nota[3] <- 6,0

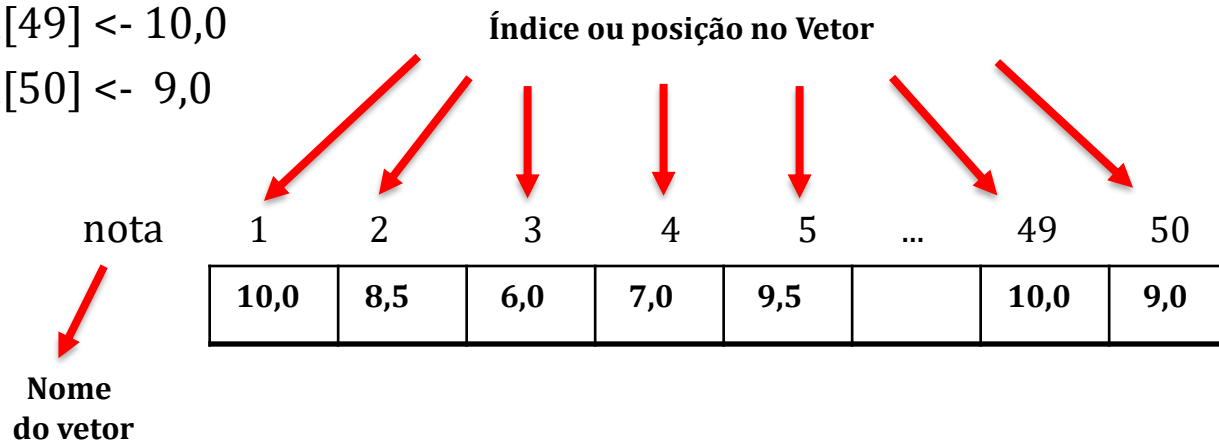
nota[4] <- 7,0

nota[5] <- 9,5

...

nota[49] <- 10,0

nota[50] <- 9,0



Atividade prática 1



Para acessar o conteúdo do vetor “Letras” é necessário determinar o nome do vetor e a posição ou índice do elemento no vetor.

Exemplo: nome[posição]

Letras	1	2	3	4	5	6	7	8	9	10
	A	B	C	D	E	F	G	H	I	J

Responda e anote as respostas:

- 1) escreval (Letras[3]) o conteúdo impresso será? **Resposta:** _____
- 2) escreval (Letras[2*2]) o conteúdo impresso será? **Resposta:** _____
- 3) Letras[6] \leftarrow Letras[2] o vetor na posição 6 recebe ? **Resposta:** _____
- 4) x \leftarrow 5
escreval (Letras[x]) o conteúdo impresso será? **Resposta:** _____

Conseguiu responder?

Agora vamos conferir



INSTITUTO FEDERAL
Rio Grande do Sul



Atividade prática 1

respostas

Letras	1	2	3	4	5	6	7	8	9	10
	A	B	C	D	E	F	G	H	I	J

- 1) escreval (Letras[3]) o conteúdo impresso será? Resposta: **C**
- 2) escreval (Letras[2*2]) o conteúdo impresso será? Resposta: **D**
- 3) Letras[6] \leftarrow Letras[2] o vetor na posição 6 recebe ? Resposta: **B**
- 4) $x \leftarrow 5$
escreval (Letras[x]) o conteúdo impresso será? Resposta: **E**

Leitura do vetor

Não se usa:

escreval ("Informe 10 valores")

leia (v[1], v[2], v[3], ..., v[9], v[10])



Forma Correta:

escreval ("Informe 10 valores")

para i de 1 ate 10 faca

leia (v[i])

fimpara

ou

para i de 1 até 10 faca

escreval ("Informe valor")

leia (v[i])

fimpara



Vamos testar no VisualG e descobrir a diferença entre as duas formas?

Forma Correta:

escreval (“Informe 10 valores”)

para i de 1 ate 10 faca

leia (v[i])

fimpara

ou

para i de 1 até 10 faca

escreval (“Informe valor”)

leia (v[i])

fimpara



Atividade prática 2



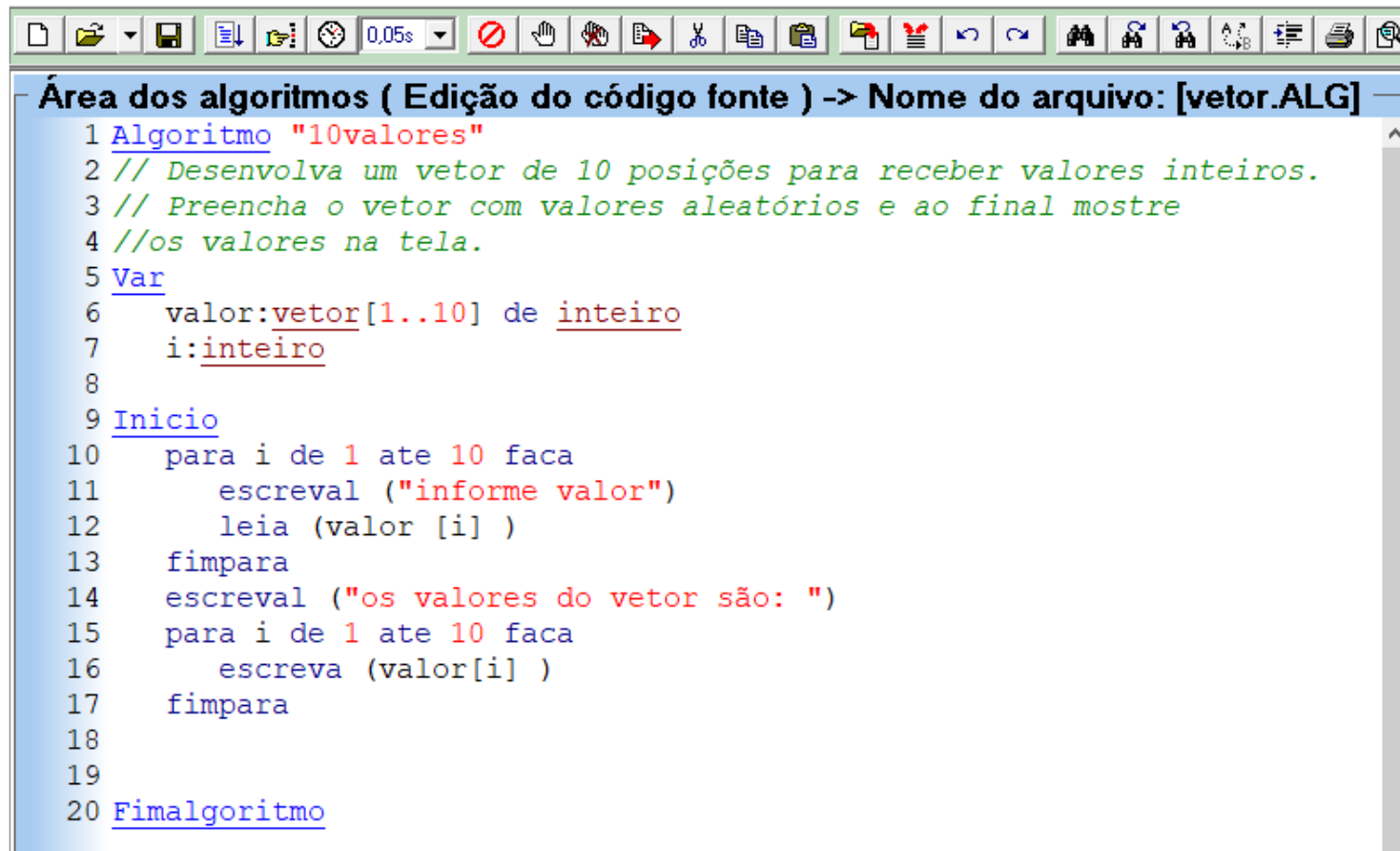
Para desenvolver no VisuAlg:

Um vetor de 10 posições para receber valores inteiros. Preencha o vetor com valores aleatórios e ao final mostre os valores na tela.



Atividade prática 2

resposta



The screenshot shows a code editor window titled "Área dos algoritmos (Edição do código fonte) -> Nome do arquivo: [vetor.ALG]". The code is written in ALG (Algebraic Language) and is as follows:

```
1 Algoritmo "10valores"  
2 // Desenvolva um vetor de 10 posições para receber valores inteiros.  
3 // Preencha o vetor com valores aleatórios e ao final mostre  
4 //os valores na tela.  
5 Var  
6   valor:vetor[1..10] de inteiro  
7   i:inteiro  
8  
9 Inicio  
10  para i de 1 ate 10 faça  
11    escreval ("informe valor")  
12    leia (valor [i] )  
13  fimpara  
14  escreval ("os valores do vetor são: ")  
15  para i de 1 ate 10 faça  
16    escreva (valor[i] )  
17  fimpara  
18  
19  
20 Fimalgoritmo
```



Atividade prática 3



Para desenvolver no VisuAlg:

Algoritmo que lê um vetor v de 5 posições e no final apresenta os valores pares do vetor.



Atividade prática 3

resposta

Área dos algoritmos (Edição do código fonte) -> Nome do arquivo: [VALORES.PAS]

```
1 Algoritmo "ValoresPares"
2 // Algoritmo que lê um vetor v de 5 posições
3 //e no final apresenta os valores pares do vetor.
4
5 Var
6   v, vp:vetor[1..5] de inteiro
7   i:inteiro
8
9 Inicio
10
11   para i de 1 ate 5 faça
12     escreval ("informe valor")
13     leia (v[i] )
14   fimpara
15   escreval ("Valores Pares")
16   para i de 1 ate 5 faça
17     se v[i] mod 2 = 0 então
18       vp[i]<- v[i]
19       escreval(vp[i])
20     fimse
21   fimpara
22
23 Fimalgoritmo
```

C:\ Console simulando o moc

```
informe valor
1
informe valor
2
informe valor
3
informe valor
4
informe valor
5
Valores Pares
 2
 4

>>> Fim da execução
```



Atividade prática 3 (modificada)



Para desenvolver no VisuAlg:

Algoritmo que lê um vetor v de 5 posições e no final apresenta **um vetor vp com os valores pares de v .**



O que muda?

Área dos algoritmos (Edição do código)

```
1 Algoritmo "ValoresPares"
2 // Algoritmo que lê um vetor v de
3 //e no final apresenta os valores
4
5 Var
6   v, vp:vetor[1..5] de inteiro
7   i:inteiro
8
9 Inicio
10
11   para i de 1 ate 5 faca
12     escreval ("informe valor")
13     leia (v[i] )
14   fimpara
15   escreval ("Valores Pares")
16   para i de 1 ate 5 faca
17     se v[i] mod 2 = 0 entao
18       vp[i]← v[i]
19       escreval(vp[i])
20     fimse
21   fimpara
22
23 Fimalgoritmo
```

Área dos programas (Edição do código)

```
1 Algoritmo "ValoresPares buraco"
2 // Algoritmo que lê um vetor v de
3 //e no final apresenta um vetor vp
4
5 Var
6   v, vp:vetor[1..5] de inteiro
7   i:inteiro
8
9 Inicio
10
11   para i de 1 ate 5 faca
12     escreval ("informe valor")
13     leia (v[i] )
14   fimpara
15   escreval ("Valores Pares")
16   para i de 1 ate 5 faca
17     se v[i] mod 2 = 0 entao
18       vp[i]← v[i]
19     fimse
20     escreval(vp[i])
21   fimpara
22
23 Fimalgoritmo
```



O que muda?

Área dos programas (Edição do código f

```
1 Algoritmo "ValoresPares buraco"
2 // Algoritmo que lê um vetor v de 5 elementos
3 //e no final apresenta um vetor vp de 5 elementos
4
5 Var
6   v, vp:vetor[1..5] de inteiro
7   i:inteiro
8
9 Inicio
10
11   para i de 1 ate 5 faca
12     escreval ("informe valor")
13     leia (v[i] )
14   fimpara
15   escreval ("Valores Pares")
16   para i de 1 ate 5 faca
17     se v[i] mod 2 = 0 entao
18       vp[i]<- v[i]
19     fimse
20     escreval(vp[i])
21   fimpara
22
23 Fimalgoritmo
```

Console simulando o mo

```
informe valor
1
informe valor
2
informe valor
3
informe valor
4
informe valor
5
Valores Pares
0
2
0
4
0

>>> Fim da execução
```



Como eliminar os buracos do vetor?

```
Área dos programas ( Edição do código f
1 Algoritmo "ValoresPares buraco"
2 // Algoritmo que lê um vetor v de 3
3 //e no final apresenta um vetor vp
4
5 Var
6   v, vp:vetor[1..5] de inteiro
7   i:inteiro
8
9 Inicio
10
11   para i de 1 ate 5 faça
12     escreval ("informe valor")
13     leia (v[i] )
14   fimpara
15   escreval ("Valores Pares")
16   para i de 1 ate 5 faça
17     se v[i] mod 2 = 0 entao
18       vp[i]<- v[i]
19     fimse
20     escreval(vp[i])
21   fimpara
22
23 Fimalgoritmo
```

```
C:\ Console simulando o mo
informe valor
1
informe valor
2
informe valor
3
informe valor
4
informe valor
5
Valores Pares
0
2
0
4
0
>>> Fim da execução
```

Vetor sem Buracos

```
1 Algoritmo "ValoresPares sem buracos"
2 // Algoritmo que lê um vetor v de 5 posições
3 //e no final apresenta um vetor vp com os valores
4
5 Var
6   v, vp:vetor[1..5] de inteiro
7   i, posp:inteiro
8
9 Inicio
10   posp <- 0
11   para i de 1 ate 5 faca
12     escreval ("informe valor")
13     leia (v[i] )
14   fimpara
15   escreval ("Valores Pares")
16   para i de 1 ate 5 faca
17     se v[i] mod 2 = 0 entao
18       posp <- posp +1
19       vp[posp]<- v[i]
20     fimse
21   fimpara
22   para i de 1 ate 5 faca
23     escreval (vp [i])
24   fimpara
25
26 Fimalgoritmo
```



Vetor sem Buracos

```
1 Algoritmo "ValoresPares sem buracos"
2 // Algoritmo que lê um vetor v de 5 posições
3 //e no final apresenta um vetor vp com os valores
4
5 Var
6   v, vp:vetor[1..5] de inteiro
7   i, posp:inteiro
8
9 Inicio
10   posp <- 0
11   para i de 1 ate 5 faca
12     escreval ("informe valor")
13     leia (v[i] )
14   fimpara
15   escreval ("Valores Pares")
16   para i de 1 ate 5 faca
17     se v[i] mod 2 = 0 entao
18       posp <- posp +1
19       vp[posp]<- v[i]
20     fimse
21   fimpara
22   para i de 1 ate 5 faca
23     escreval (vp [i])
24   fimpara
25
26 Fimalgoritmo
```

Console simulando o mo

```
informe valor
1
informe valor
2
informe valor
3
informe valor
4
informe valor
5
Valores Pares
2
4
0
0
0

>>> Fim da execução
```



Vetor sem Buracos (aprimorado)

Área dos programas (Edição do código fonte)

```
1 Algoritmo "ValoresPares sem buracos"
2 // Algoritmo que lê um vetor v de 5 posi
3 //e no final apresenta um vetor vp com o
4
5 Var
6   v, vp:vetor[1..5] de inteiro
7   i, posp:inteiro
8
9 Inicio
10   posp <- 0
11   para i de 1 ate 5 faca
12     escreval ("informe valor")
13     leia (v[i] )
14   fimpara
15   escreval ("Valores Pares")
16   para i de 1 ate 5 faca
17     se v[i] mod 2 = 0 entao
18       posp <- posp +1
19       vp[posp]<- v[i]
20     fimse
21   fimpara
22   para i de 1 ate posp faca
23     escreval (vp [i])
24   fimpara
25
26 Fimalgoritmo
```

Console simulando o mo

```
informe valor
1
informe valor
2
informe valor
3
informe valor
4
informe valor
5
Valores Pares
2
4
>>> Fim da execução
```



Atividade prática 4



Para desenvolver no VisuAlg e enviar pelo link no moodle:

Algoritmo que leia a altura e o nome dos 50 jogadores do time de basquete do IFRS. Ao final o programa deve apresentar na tela:

- a) A maior altura;
- b) A menor altura;
- c) A média de altura dos jogadores.



Atividade prática 4 (modificada)



E se o programa tiver de imprimir na tela o **nome do jogador** que tem a **maior altura** e o **nome** do que tem a **menor altura**?



Atividade prática 5



Criar um algoritmo que leia 10 números pelo teclado e exiba os números na ordem inversa da que os números foram digitados.



Atividade prática 5

resposta

```
algoritmo "vetor"  
var  
  numeros: vetor [1..10] de inteiro  
  i: inteiro  
inicio  
  para i de 1 ate 10 faca  
    escreva("Digite um número ")  
    leia(numeros[i])  
  fimpara  
  escreva("Numeros na ordem inversa: ")  
  para i de 10 ate 1 passo -1 faca  
    escreva(numeros[i])  
  fimpara  
fimalgoritmo
```



Atividade prática 6



Escreva um algoritmo que leia um vetor com 10 posições de números inteiros. Em seguida, receba um novo valor do usuário e verifique se este valor está contido no vetor.



Atividade prática 6 (resposta)

Área dos algoritmos (Edição do código fonte) -> Nome do arquivo: [semnom]

```
1 Algoritmo "Busca_valor"  
2 //Escreva um algoritmo que leia um vetor com 10 posições de números  
3 //inteiros. Em seguida, receba um novo valor do usuário e verifique  
4 //se este valor está contido no vetor.  
5 Var  
6   numeros:vetor[1..10] de inteiro  
7   i, valor:inteiro  
8   encontrou:logico  
9  
10 Inicio  
11   para i de 1 ate 10 faça  
12     escreval ("Digite um número")  
13     leia (numeros[i])  
14   fimpara  
15   escreval ("Digite um valor p busca no vetor")  
16   leia (valor)  
17   encontrou <- falso  
18   para i de 1 ate 10 faça  
19     se numeros[i] = valor entao  
20       encontrou := verdadeiro  
21     fimse  
22   fimpara  
23   se encontrou = verdadeiro entao  
24     escreval("O valor está contido no vetor")  
25   senão  
26     escreval("O valor não está contido no vetor")  
27   fimse  
28 Fimalgoritmo
```



Curiosidade

Area dos algoritmos (Edição do código fonte) -> Nome do arquivo

```
1 Algoritmo "Busca_valor"
2 //Escreva um algoritmo que leia um vetor com 10 posições
3 //inteiros. Em seguida, receba um novo valor do usuário e
4 //se este valor está contido no vetor.
5 Var
6   numeros:vetor[1..10] de inteiro
7   i, valor:inteiro
8   encontrou:logico
9
10 Inicio
11   para i de 1 ate 10 faça
12     escreval ("Digite um número")
13     leia (numeros[i])
14   fimpara
15   escreval ("Digite um valor p busca no vetor")
16   leia (valor)
17   encontrou <- falso
18   para i de 1 ate 10 faça
19     se numeros[i] = valor entao
20       encontrou := verdadeiro
21       interrompa
22     fimse
23   fimpara
24   se encontrou = verdadeiro entao
25     escreval("O valor está contido no vetor")
26   senão
27     escreval("O valor não está contido no vetor")
28   fimse
29 Fimalgoritmo
```

As estruturas de repetição
(Para, Enquanto e Repita)
permitem o uso do comando

INTERROMPA

Esse comando possibilita a
saída imediata do laço de
repetição

Ao encontrar esse
comando, o
algoritmo passa a
execução para o
próximo comando
após o laço

VIDA DE PROGRAMADOR COM.BR

```
real historia;  
string sender;  
sender = "Carlos";
```



#1211

ENTÃO, ALONSO, É ASSIM QUE FUNCIONA O FOR EM C... AGORA DECLARA AÍ UM VETOR COM 4 POSIÇÕES E INICIALIZA OS VALORES DE 1 A 4...

```
(...)  
V[0] = 1;  
V[1] = 2;  
V[2] = 3;  
V[3] = 4;
```



AFF, ALONSO! E SE O VETOR TIVER 100 POSIÇÕES?

COPIA E COLA...



E SE TIVER UM MILHÃO DE POSIÇÕES...

HMMMM... AÍ COPIA E COLA BASTANTE...

KABOOM!

