



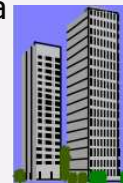
## Algoritmos e Programação – Parte 09

Prof. Jorge Cavalcanti  
[jorge.cavalcanti@univasf.edu.br](mailto:jorge.cavalcanti@univasf.edu.br)  
[www.univasf.edu.br/~jorge.cavalcanti](http://www.univasf.edu.br/~jorge.cavalcanti)  
[www.twitter.com/jorgecav](http://www.twitter.com/jorgecav)

## Algoritmos e Programação

### Vetores

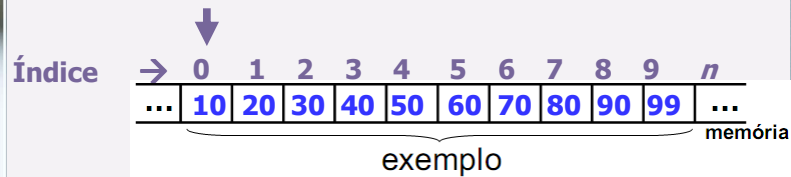
- Estrutura de Dados Homogênea e Estática
  - Unidimensional
- Exemplo :
  - Prédio com **um** apartamento por andar
  - Conjunto habitacional com apenas uma rua
- Todos os elementos pertencentes ao mesmo tipo de dado;



## Algoritmos e Programação

### Vetores

- Índices (iniciam em "0", até "n");
- Índices utilizados para Recuperar/Inserir valores.



- Forma geral para se declarar um vetor:
- **tipo\_da\_variável nome\_da\_variável [tamanho];**

3

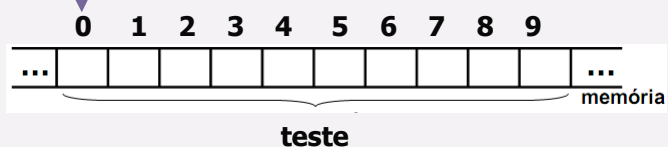
## Algoritmos e Programação

### Vetores

#### Exemplo:

- float teste [10].

↓ Todos os valores são do tipo float



- Em C a numeração dos índices começa sempre em zero. Isto significa que, no exemplo acima, os dados serão indexados de 0 a 9. Para acessá-los vamos escrever: **teste [0]..... teste [9]**

4

## Algoritmos e Programação

### Vetores

- Mas ninguém o impede (programador) de escrever:
  - teste[30]
  - teste[103]
  - teste[-2]
- O C não verifica se o índice que você usou está dentro dos limites válidos. Este é um cuidado que você deve tomar.
- Se o programador não tiver atenção com os limites de validade para os índices ele corre o risco de ter variáveis sobrescritas ou de ver o computador travar. Inúmeros bugs podem surgir.

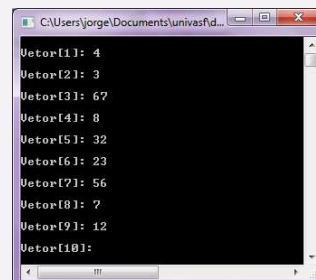
5

## Algoritmos e Programação

### Vetores

- Exercício 1:  
Construa um programa que declare um vetor de inteiros com 10 elementos e o inicialize com números fornecidos pelo usuário, através da entrada padrão.

```
#include <stdio.h>
main()
{
    int vetor[10], indice;
    for (indice=0; indice<10; indice++)
    {
        printf("\nVetor[%d]: ", indice);
        scanf("%d", &vetor[indice]);
    }
}
```



```
C:\Users\jorge\Documents\univasfd...
Vetor[1]: 4
Vetor[2]: 3
Vetor[3]: 67
Vetor[4]: 8
Vetor[5]: 32
Vetor[6]: 23
Vetor[7]: 56
Vetor[8]: 7
Vetor[9]: 12
Vetor[10]:
```

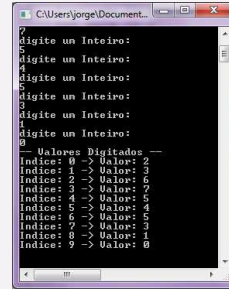
6

## Algoritmos e Programação

### ■ Exercício 2:

Construa um programa que declare e receba um vetor de inteiros com 10 elementos com números fornecidos pelo usuário, através da entrada padrão e depois exiba os índices e seus valores armazenados.

```
#include <stdio.h>
int main () {
int vetorInteiros[10];
for(int x=0; x<10; x++){
printf("digite um Inteiro: \n");
scanf("%d",&vetorInteiros[x]);
if(x==9){
printf("-- Valores Digitados -- \n");
for(int y=0; y<10; y++){
printf("Indice: %d -> Valor: %d \n", y, vetorInteiros[y]); }
} }
getchar();
return(0); }
```



```
C:\Users\jorge\Document...
digite um Inteiro:
2
digite um Inteiro:
3
digite um Inteiro:
6
digite um Inteiro:
7
digite um Inteiro:
5
digite um Inteiro:
4
digite um Inteiro:
5
digite um Inteiro:
3
digite um Inteiro:
1
digite um Inteiro:
0
-- Valores Digitados --
Indice: 0 -> Valor: 2
Indice: 1 -> Valor: 3
Indice: 2 -> Valor: 6
Indice: 3 -> Valor: 7
Indice: 4 -> Valor: 5
Indice: 5 -> Valor: 4
Indice: 6 -> Valor: 5
Indice: 7 -> Valor: 3
Indice: 8 -> Valor: 1
Indice: 9 -> Valor: 0
```

7

## Algoritmos e Programação

### Vetores

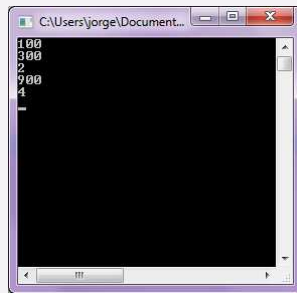
- Um vetor pode ser inicializado na declaração, exemplo:
  - `int vetor[10]={0,1,2,3,4,5,6,7,8,9};`
- E ainda pode-se deixar em aberto o número de elementos, que será preenchido pelo números de elementos na inicialização, que ocorre no momento da declaração. Ou seja:
  - `int vetor[]={0,1,2,3,4,5,6,7,8,9};`
- E não:
  - `int vetor[];`

8

## Algoritmos e Programação

### Vetores

- Valores das posições podem ser modificados no programa:



```
#include <stdio.h>
int main() {
    int sal[]={0,1,2,3,4};
    sal[0]=100;
    sal[1]=300;
    sal[3]=900;
    printf ("%d\n",sal[0]);
    printf ("%d\n",sal[1]);
    printf ("%d\n",sal[2]);
    printf ("%d\n",sal[3]);
    printf ("%d\n",sal[4]);
    getchar();
    return (0); }
```

9

## Algoritmos e Programação

### Matrizes

- Vetores Multidimensionais
- Estrutura de Dados Homogênea
  - Bidimensional
- Exemplo :
  - Prédio com mais de um apartamento por andar
  - Conjunto habitacional com várias ruas



10

# Algoritmos e Programação

## Matrizes

- Declarando uma matriz  
tipo\_da\_variável nome\_da\_variável [d1][d2]...[dn];

- Exemplo:
- Declarar uma estrutura com o nome de salario, com dez elementos inteiros dispostos numa matriz com cinco linhas e duas colunas.

```
int salario[5][2];
```

4		
3		
2		
1		
0		
	0	1

11

# Algoritmos e Programação

## Matrizes

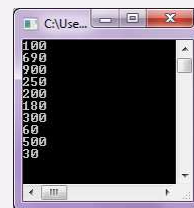
- Exemplo:

```
#include <stdio.h>
main(){
int sal[5][2],x,y;
sal[0][0]=100;
sal[0][1]=690;
sal[1][0]=900;
sal[1][1]=250;
sal[2][0]=200;
sal[2][1]=180;
sal[3][0]=300;
sal[3][1]=60;
sal[4][0]=500;
sal[4][1]=30;
```

```
for(x=0;x<5;x++){
for(int y=0;y<2;y++){
printf ("%d\n", sal[x][y]);}}
```

```
getchar();
return(0);
}
```

4	500	30
3	300	60
2	200	180
1	900	250
0	100	690
	0	1



12

## Algoritmos e Programação

- Exemplo: O programa abaixo cria e exibe uma matriz com dimensões e valores definidas pelo usuário .

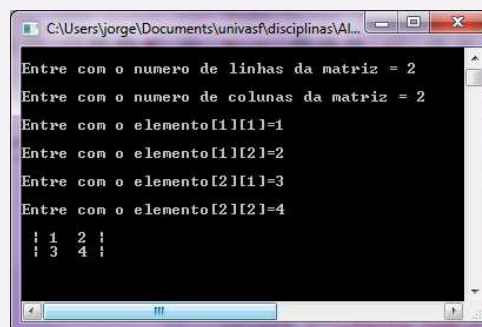
```
#include <stdio.h>
int main()
{
    // definição do tamanho da matriz
    int nl, nc;
    printf ("\nEntre com o numero de linhas da matriz = ");
    scanf ("%d",&nl);
    printf ("\nEntre com o numero de colunas da matriz = ");
    scanf ("%d",&nc);
    // entrada de dados da matriz
    int matriz[nl][nc], i, j;
    for (i=0;i<nl;i++)
        for (j=0;j<nc;j++)
        {
            printf ("\nEntre com o elemento[%d][%d]=",i+1,j+1);
            scanf ("%d",&matriz[i][j]); }
}
```

13

## Algoritmos e Programação

- Continuação Exemplo:

```
// impressão da matriz na tela
for (i=0;i<nl;i++){
    printf("\n |"); // barra vertical
    for (j=0;j<nc;j++)
        printf (" %d ",matriz[i][j]);
    printf("|");
}
getchar();
return (0);
}
```



```
CAUsers\jorge\Documents\univasf\disciplinas\Al...
Entre com o numero de linhas da matriz = 2
Entre com o numero de colunas da matriz = 2
Entre com o elemento[1][1]=1
Entre com o elemento[1][2]=2
Entre com o elemento[2][1]=3
Entre com o elemento[2][2]=4

| 1 2 |
| 3 4 |
```

14

## Algoritmos e Programação

### Matrizes

- Assim como os vetores unidimensionais os vetores multidimensionais também podem ser inicializados na declaração.

- Exemplo:

```
float matriz [3][4]={1,2,3,4,5,6,7,8,9,10,11,12};
```

```
int matriz [ ][2]={1,2,3,4,5,6,7,8,9,10,11,12};
```

- Não use:

```
float matriz [ ][ ]={1,2,3,4,5,6,7,8,9,10,11,12};
```

O Exemplo a seguir usa exemplo de matriz dimensionada e não-dimensionada.

15

## Algoritmos e Programação

### Matrizes

```
#include<stdio.h>
```

```
int main() {
```

```
int mat1[2][2]={4,5,-2,1};
```

```
int mat2[][2]={4,5,-2,1};
```

```
/*Matriz não-dimensionada*/
```

```
int x,y;
```

```
printf("Imprimindo a matriz mat1  
cujo o tamanho foi especificado:\n");
```

```
for(x=0;x<2;x++){
```

```
for(y=0;y<2;y++){
```

```
printf("mat1[%d][%d]=%d",x+1,y+  
1,mat1[x][y]);
```

```
printf("\n"); } /*fim do for*/ } /*fim  
do for*/
```

```
printf("\n");
```

```
printf("Imprimindo a matriz  
mat2 cujo o tamanho nao  
foi especificado:\n");
```

```
for(x=0;x<2;x++){
```

```
for(y=0;y<2;y++){
```

```
printf("mat2[%d][%d]=
```

```
%d",x+1,y+1,mat2[x][y]);
```

```
printf("\n");
```

```
} /*fim do for*/
```

```
} /*fim do for*/
```

```
getchar();
```

```
return(0);} /*fim do programa*/
```

16



# Algoritmos e Programação

## Matrizes

```
C:\Users\jorge\Documents\univasf\disciplinas\Algoritmos\codigos_fontes\teste_matriz_dimensic
Imprimindo a matriz mat1 cujo o tamanho foi especificado:
mat1[0][0]=4
mat1[0][1]=5
mat1[1][0]=-2
mat1[1][1]=1

Imprimindo a matriz mat2 cujo o tamanho nao foi especificado:
mat2[0][0]= 4
mat2[0][1]= 5
mat2[1][0]= -2
mat2[1][1]= 1
```

17

# Algoritmos e Programação

## Matrizes

- Exercício 01 - Faça um programa que exibe o conteúdo de um vetor tamanho 10 de forma invertida.
- Exercício 02 - Faça um programa que leia um vetor de 10 posições e crie um segundo vetor substituindo os valores negativos por 1.
- Exercício 03 - Faça um programa que leia uma matriz **mat** 3 x 4 de inteiros, substitua seus elementos negativos por 0 e imprima a matriz **mat** original e a modificada.

18

## Algoritmos e Programação

- Exercício 01 - Faça um programa que exibe o conteúdo de um vetor tamanho 10 de forma invertida.

```
#include <stdio.h>
#include <conio.h>

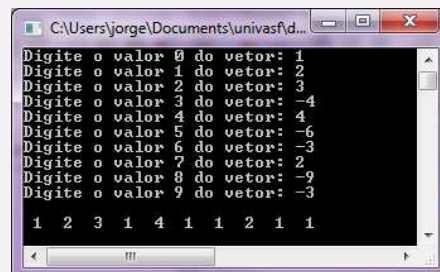
int main () {
    int vet1[10];
    for (int x=0; x<10;x++)
    {
        printf ("Indice: %d - Digite um inteiro:", x);
        scanf ("%d", &vet1[x]);
        if (x==9){
            printf("\n \n -- Valores digitados --\n \n Vetor 2(Invertido)\n \n");
            for (int y=9; y>=0; y--){
                printf ("Indice: %d - Valor %d \n", y, vet1[y]);
            }
        }
    }
    getch();
    return(0);
}
```

19

## Algoritmos e Programação

- Exercício 02 - Faça um programa que leia um vetor de 10 posições e crie um segundo vetor substituindo os valores negativos por 1.

```
#include <stdio.h>
#include <conio.h>
int main () {
    int vet[10];
    int i=0;
    do {
        printf ("Digite o valor %d do vetor: ", i);
        scanf ("%d", &vet[i]);
        i++; }
    while (i<=9);
    printf ("\n");
    for (i=0; i<10; i++){
        if (vet[i]<0){
            vet[i]=1;
            printf(" %d ", vet[i]);
        }
    }
    getch();
    return(0);
}
```



```
C:\Users\jorge\Documents\univas\d...
Digite o valor 0 do vetor: 1
Digite o valor 1 do vetor: 2
Digite o valor 2 do vetor: 3
Digite o valor 3 do vetor: -4
Digite o valor 4 do vetor: -4
Digite o valor 5 do vetor: -6
Digite o valor 6 do vetor: -3
Digite o valor 7 do vetor: 2
Digite o valor 8 do vetor: -9
Digite o valor 9 do vetor: -3

1 2 3 1 4 1 1 2 1 1
```

20

## Algoritmos e Programação

- Exercício 03 - Faça um programa que leia uma matriz **mat** 3 x 4 de inteiros, substitua seus elementos negativos por 0 e imprima a matriz **mat** original e a modificada.

```
#include <stdio.h>
#include <conio.h>
main() {
int matriz[3][4], i, j;
for (i=0;i<3;i++) {
for (j=0;j<4;j++) {
printf ("\nEntre com o elemento[%d][%d]=",i+1,j+1);
scanf ("%d",&matriz[i][j]); } }
printf ("\n Matriz Principal \n");
for (i=0;i<3;i++) {
for (j=0;j<4;j++) {
printf ("\n Valor: %d",matriz[i][j]); }}
printf ("\n \n Matriz Modificada \n");
for (i=0;i<3;i++) {
if (matriz [i][j]<0) {
matriz[i][j]=0;}
printf ("\n Valor: %d",matriz[i][j]); }
getch();
return (0);
}
}
```

21