



Linguagem C – Vetores e Matrizes

Curso: Ciência da Computação
Disciplina: Algoritmos e Programação
Prof. Luciano Antunes



Estruturas Homogêneas

Vetores e matrizes

- Uma matriz é uma coleção de variáveis de mesmo tipo, acessíveis com um único nome e armazenados contiguamente na memória.
- A individualização de cada variável de um vetor é feita através do uso de índices.
- Os Vetores são matrizes de uma só dimensão.



Estruturas Homogêneas

Vetores

- Vetor é uma variável composta homogênea unidimensional, formada por uma seqüência de variáveis, todas do mesmo tipo, com o mesmo identificador (mesmo nome) e alocadas seqüencialmente na memória principal (um após o outro).
- Os vetores são utilizados quando se tem vários valores de um mesmo tipo, sendo melhor agrupá-los do que trabalhar com eles individualmente.
- O número de elementos de um vetor é fixo e determinado no momento de sua declaração.



Estruturas Homogêneas

Vetores

- Exemplo:

Vetor A

10	15	20	12	35	28	33	67
0	1	2	3	4	5	6	7

- Esse é um vetor de 8 elementos, isto é, tem 8 variáveis, todas com o mesmo nome e diferentes por sua posição dentro do arranjo que é indicada por um **índice**.

$A[0] = 10$ $A[1] = 15$ $A[2] = 20$ $A[3] = 12$ $A[4] = 35$ $A[5] = 28$ $A[6] = 33$ $A[7] = 67$



Estruturas Homogêneas

Vetores

DECLARAÇÃO – SINTAXE

tipo nome [dimensão];

- **Sendo:**

tipo: é o tipo básico de dados que poderá ser armazenado na seqüência de variáveis que formam o vetor. Poderá ser **int**, **float** ou **char**.

nome: é o dado ao vetor dentro das regras para nomear uma variável.

[dimensão]: é o número de elementos, ou seja, a quantidade de variáveis que vão formar o vetor.

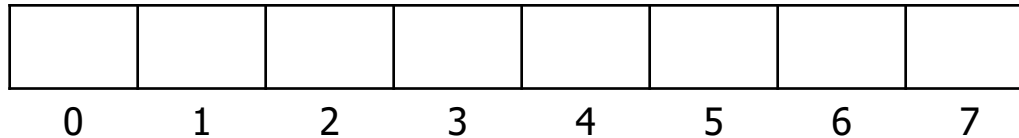


Estruturas Homogêneas

Vetores

Exemplos:

int A[8]; // 8 elementos, variando de 0 a 7





Estruturas Homogêneas

Vetores - Atribuição de valores ao vetor

$A[0] = 10;$

10							
0	1	2	3	4	5	6	7

$A[1] = 15;$

10	15						
0	1	2	3	4	5	6	7

$A[4] = 35;$

10	15			35			
0	1	2	3	4	5	6	7



Estruturas Homogêneas

Vetores -

Fases de utilização de um vetor em um algoritmo :

Dimensionamento:

tipo nome [dimensão];

Ex: **float** salario[5];

Vetor **salario**

100,00	150,50	220,30	350,80	122,00
0	1	2	3	4



Estruturas Homogêneas

Vetores -

```
int idade[3];
```

Vetor **idade**

45	22	15
0	1	2

```
char nome[6][20];
```

Vetor **nome**

Carlos	Pedro	Maria	João	José	Tereza
0	1	2	3	4	5



Estruturas Homogêneas

Vetores – Entrada de Dados

- Normalmente, faz uso de uma estrutura de repetição.
- Se for a estrutura do **for**, deverá ter o valor final igual a última posição do vetor.
- Se for a estrutura do **while**, deverá ter uma variável que será incrementada e nunca poderá assumir um valor maior do que a última posição do vetor.



Estruturas Homogêneas

Vetores – Entrada de Dados

```
for (i=0; i<=4; i++)  
{  
    printf("Entre o valor do salário ");  
    scanf("%f", &salario[i]);  
}
```

```
for (i=0; i<=5; i++)  
{  
    printf("Entre com o nome da pessoa");  
    scanf("%s", &nome[i]);  
}
```

```
for (i=0; i<=2; i++)  
{  
    printf("Entre a idade da pessoa");  
    scanf("%d", &idade[i]);  
}
```



Estruturas Homogêneas

Vetores – Saída de Dados

- Normalmente, faz uso de um estrutura de repetição.
- Se for a estrutura do **for**, deverá ter o valor final igual a última posição do vetor.
- Se for a estrutura do **while**, deverá ter uma variável que será incrementada e nunca poderá assumir um valor maior do que a última posição do vetor.



Estruturas Homogêneas

Vetores – Saída de Dados

```
for (i=0; i<=4; i++)  
{  
    printf("O valor do salário é: %d", salario[i]);  
}
```

```
for (i=0; i<=5; i++)  
{  
    printf("O nome da pessoa é: %s", nome[i]);  
}
```

```
for (i=0; i<=2; i++)  
{  
    printf("A idade da pessoa é: %s", idade[i]);  
}
```

Estruturas Homogêneas - Matrizes



- A declaração de uma matriz bidimensional é muito parecida com a declaração de um vetor:

```
tipo_da_variável nome_da_variável [altura][largura];
```

- O índice da esquerda indexa as linhas e o da direita indexa as colunas.
- Mais uma vez é bom lembrar que:
 - Os índices variam de zero ao valor declarado menos um;
 - O C não verifica a faixa de índices válidos para o desenvolvedor

Estruturas Homogêneas - Matrizes

■ Exemplo:

```
#include <stdio.h>
int main ()
{
    int matriz [20][10];
    int i,j,cont=1;
    //Populando a matriz
    for (i=0;i<20;i++)
    {
        for (j=0;j<10;j++)
        {
            matriz[i][j]=cont;
            cont++;
        }
    }
}
```

```
//Leitura da matriz formatada
for (i=0;i<20;i++)
{
    for (j=0;j<10;j++)
    {
        printf("%d\t",
matriz[i][j]);
    }
    printf("\n");
}

return (0);
}
```

Inicialização Vetor/Matriz

- Pode-se inicializar matrizes, assim como pode-se inicializar variáveis:

```
float vect [6] = { 1.3, 4.5, 2.7, 4.1, 0.0, 100.1 };  
int matriz [3][4] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 };
```

- Demonstra a inicialização de matrizes multidimensionais, onde a matriz está sendo inicializada com
 - 1, 2, 3 e 4 em sua primeira linha,
 - 5, 6, 7 e 8 na segunda linha
 - 9, 10, 11 e 12 na última linha.
- Inicialização sem especificação de tamanho:

```
int matrxx [][] = { 1,2,2,4,3,6,4,8,5,10 };
```

- No exemplo acima o valor não especificado será 5.