



ESTRUTURA DE DADOS E ALGORITMOS I

Estruturas Homogêneas -

Vetores

Profº. Sérgio Roberto Costa Vieira, M.Sc.

Cursos de Computação

2º. Período

Estrutura de Dados Homogêneas

Introdução

Um dos objetivos da criação de um programa é a manipulação dos dados.

As estruturas de dados consistem em arranjos ou organizações lógicas dos dados de um programa.

A organização dos dados é denominada de estrutura de dados composta e possui duas classificações:

- Estruturas de Dados Homogêneas ←
- Estruturas de Dados Heterogêneas



Estrutura de Dados Homogêneas

Introdução

As estruturas de dados homogêneas representam um conjunto de valores do mesmo tipo de dados (por exemplo: char, int, float etc.).

Possibilitando o armazenamento de grupos ou conjunto de valores em uma única variável, que será armazenada na memória do computador.

Esses dados são individualizados entre si através da posição que ocupam nesse conjunto (variáveis indexadas).

Estrutura de Dados Homogêneas

Introdução

O termo **indexada** refere-se a maneira de como é feita a individualização dos elementos no conjunto de dados.

Cada dado possui a sua posição e é identificado através de um índice.



Uma variável indexada pode ser definida como tendo um ou mais índices, sendo classificada como:

- Variáveis unidimensionais (vetores) ←
- Variáveis multidimensionais (matrizes)

Estrutura de Dados Homogêneas

Vetores

DEFINIÇÃO

Um vetor (*array*) é uma estrutura de dados que pode ser acessada de forma aleatória e seus elementos são acessados por indexação, onde a variável de **indexação (índice)** é uma **variável inteira** que inicia sempre em 0 (zero).

Vetores também são conhecidos como **variável composta homogênea unidimensional**.

Vetores nada mais são que matrizes unidimensionais.

Nome do array (todos os elementos tem o mesmo nome)

↓

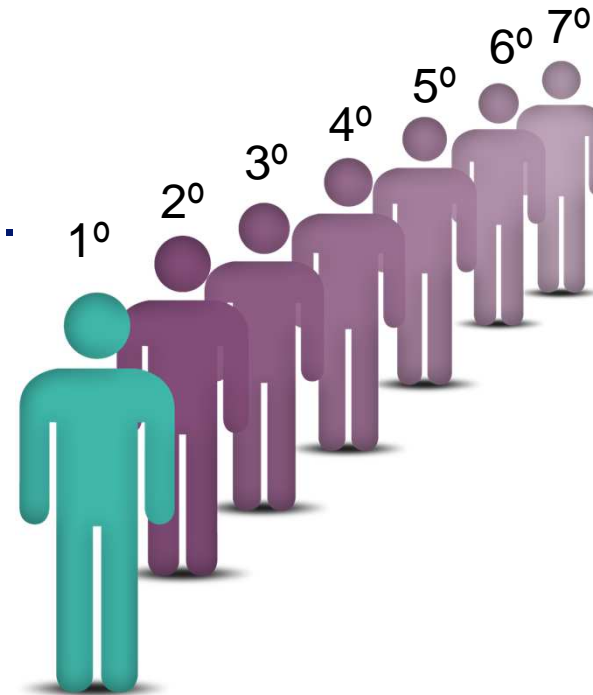
V[0]	-45
V[1]	6
V[2]	0
V[3]	-89
V[4]	62
V[5]	6453
V[6]	78
V[7]	1543

Estrutura de Dados Homogêneas

Vetores

O QUE É SER INDEXADO

Imagine uma fila de espera...



Cada pessoa na fila possui uma posição.

A posição indica a pessoa indexada na fila.

Estrutura de Dados Homogêneas

Vetores

Isso quer dizer que se trata de um conjunto de **variáveis** do mesmo tipo, que possuem o mesmo identificador (**nome**) e são alocadas sequencialmente na memória.

Como têm o mesmo nome, o que distingue é um **índice** que referencia sua localização dentro da estrutura.

Exemplo de Vetor:



Estrutura de Dados Homogêneas

Vetores

DECLARAÇÃO

Os vetores são identificados pela existência de um conjunto de colchetes ([]) logo após o nome da variável no momento da declaração.

Dentro dos colchetes deve colocar o número da quantidade de elementos que o vetor poderá armazenar, com um valor inteiro fixo.

Sintaxe: tipo_de_dados nome_da_variável [tamanho];

Exemplo: int X[8];
 float Y[7]

Estrutura de Dados Homogêneas

Vetores

DECLARAÇÃO

`tipo_dados nome_da_variável [tamanho];`

– Onde:

`tipo_dados` → refere-se ao tipo de dados (char, int, float etc);

`nome_da_variável` → identifica o nome do vetor;

`[tamanho]` → é a quantidade de elementos que pode armazenar;

Exemplo

`int numeros[10]` → seu índice varia de 0 a 9

`char data[12]` → seu índice varia de 0 a 11

Estrutura de Dados Homogêneas

Exemplo de Vetores

`int vet[5];`

vet	19	44	21	7	10
	0	1	2	3	4

`float X[6];`

X	1.5	2.8	21.3	42.7	7.9	61.3
	0	1	2	3	4	5

`char sr[6];`

sr	A	*	2	@	k	\0
	0	1	2	3	4	5

Estrutura de Dados Homogêneas

Vetores

Um outro exemplo de declaração seria definindo um valor constante para representar o tamanho do vetor utilizando a palavra reservada `#define`.

É importante ressaltar que na Linguagem C, não existe o tipo de dado *string* para armazenar uma cadeia de caracteres, deve-se declarar um vetor do tipo *char*.

Exemplo:

```
#define TAM 6;  
char Z[TAM];
```

Z	M	A	R	I	A	\0
	0	1	2	3	4	5

Estrutura de Dados Homogêneas

Vetores

ATRIBUINDO VALORES

As atribuições em vetor exigem que seja informada em qual de suas posições o valor ficará armazenado.

Lembre-se: sempre a primeira posição de um vetor tem o índice inicial de valor 0 (zero).

`vet [0] = 1 ;`

`x [3] = 'b' ;`

`sr [4] = 6.8 ;`

Estrutura de Dados Homogêneas

Vetores

Vale salientar a importância de **inicializar um vetor**, garantindo que o mesmo possua valores determinados.

Normalmente, para seu preenchimento, utiliza-se **estrutura de repetição** para tal tarefa.

Mas, também é possível inicializar o vetor no momento de sua declaração.

```
int idade[6] = {32, 16, 10, 35, 5, 18};
```

```
float notas[4] = {5.5, 9.0, 6.3, 8.5};
```

```
char nome[5] = "maria";
```


Estrutura de Dados Homogêneas

Vetores

ENTRADA DE DADOS

Preencher um vetor significa atribuir valores às suas posições. Assim, deve-se implementar um mecanismo que controle o valor do índice.

```
float notas[4];  
  
for ( i = 0; i < 4; i++ ) { //Não esqueça que o índice do vetor  
                           inicia em 0 (zero)  
    printf(" \n Digite a nota do aluno: ");  
    scanf(" %f ", &notas[ i ]); //a variável i indica o índice  
                                do vetor  
}
```

Estrutura de Dados Homogêneas

Vetores

SAÍDA DE DADOS

Para mostrar o valores contidos em um vetor também exige a utilização de um índice.

```
for ( i = 0; i < 4; i++ ) { //Não esqueça que o índice do vetor  
                           inicia em 0 (zero)  
    printf(" \n Nota: %.2f ", notas[ i ]);  
}
```

Estrutura de Dados Homogêneas

Exemplo de Vetores

```
1  #include<stdio.h> //biblioteca de comandos de entrada e saída
2  #include<conio.h> //biblioteca do comando getch()
3
4  int main(){ //função principal que inicia a execução
5
6      int n[ 8 ]; //n é o nome do array com 8 elementos inteiros
7      int i; //a variável i é uma contadora
8
9      //preenchendo o array
10     for(i = 0; i < 8; i++){ //laço de repetição usado para percorrer o array
11         n[ i ] = i + 3; //define o elemento de cada posição do array
12     } // fim do for
13
14     printf("%s%7s \n", "Elemento", "Valor");
15
16     //imprimindo o array
17     for(i = 0; i < 8; i++){ //laço de repetição usado para percorrer o array
18         printf(" n[%d]=%7d \n", i, n[ i ]); //mostra o valor de cada posição
19     } // fim do for
20
21     getch(); //comando para pausar a execução e aguardar uma tecla
22     return 0; //indica conclusão do programa e retorno da função main
23 }
```

Estrutura de Dados Homogêneas

Exemplo de Vetores

```
1 //Usando vetores: Salário Mensal - 12 meses
2 #include<stdio.h>
3 #include<stdlib.h>
4 int main(){
5     float sal_mes[12];
6     int i;
7
8     for(i=0; i<12; i++ ){ //inicializando o vetor
9         sal_mes[i]=0;
10    }
11    for(i=0; i<12; i++){ //entrada de dados
12        printf("Informe o valor do %d%c salario: R$ ", i+1, 167);
13        scanf("%f", &sal_mes[i]);
14    }
15    system("pause");
16    for(i=0; i<12; i++){ // saída de dados
17        printf("\n %do mes equivale a: R$ %.2f", i+1, sal_mes[i]);
18    }
19    printf("\n");
20    system("pause");
21    return 0;
22 }
```

Estrutura de Dados Homogêneas

Exemplo de Vetores

Este exemplo mostra o uso de vetor
sendo inicializado no momento da
declaração

```
1  #include<stdio.h>
2  #include<conio.h>
3
4  int main(){
5
6      int n[ 8 ] = {32, 27, 64, 18, 95, 37, 14, 70};
7      int i;
8
9      printf("%s%7s \n", "Elemento", "Valor");
10
11     for(i = 0; i < 8; i++){
12         printf(" n[%d]=%7d \n", i, n[ i ]);
13     }
14
15     getch();
16     return 0;
17 }
```


Estrutura de Dados Homogêneas

Vetores

POR QUE USAR?

Imagine o seguinte problema:

Você deve calcular a média aritmética das notas de prova de cinco alunos:

Dependendo do problema, o número de variáveis pode crescer bastante e o gerenciamento de variáveis comuns é complicado:

- Prováveis problemas:
 - ✓ Necessidade de criar um nome para cada variável
 - ✓ Necessidade de inicializar as variáveis corretamente
 - ✓ Dificuldade de lembrar do nome de cada uma das variáveis criadas
 - ✓ Além de ocupar vários espaços na memória

Estrutura de Dados Homogêneas

Exemplo de Vetores (sem vetor)

```
1  /* Calcula a média de cinco notas (não usa vetor) */
2  #include<stdio.h>
3  #include<conio.h>
4  int main(){
5      int nota1, nota2, nota3, nota4, nota5;
6      float media=0.0;
7
8      printf("\n Digite a nota do aluno 1: ");
9      scanf("%d", &nota1);
10     printf("\n Digite a nota do aluno 2: ");
11     scanf("%d", &nota2);
12     printf("\n Digite a nota do aluno 3: ");
13     scanf("%d", &nota3);
14     printf("\n Digite a nota do aluno 4: ");
15     scanf("%d", &nota4);
16     printf("\n Digite a nota do aluno 5: ");
17     scanf("%d", &nota5);
18
19     media = (nota1 + nota2 + nota3 + nota4 + nota5) / 5.0;
20     printf("\n Média das notas: %.2f ", media);
21
22     getch();
23     return 0;
24 }
```

Este exemplo não
usa vetor

Estrutura de Dados Homogêneas

Exemplo de Vetores

```
1  /* Calcula a média de cinco notas (usa vetor) */
2  #include<stdio.h>
3  #include<conio.h>
4  int main(){
5      int notas[5];
6      int i;
7      float media=0.0;
8
9      for(i=0; i < 5; i++){
10         printf("\n Digite a nota do aluno %d: ", i+1);
11         scanf("%d", &notas[i]);
12         media += notas[i];
13     }
14
15     media = media / 5.0;
16     printf("\n Média das notas: %.2f ", media);
17
18     getch();
19     return 0;
20 }
21
```

Este exemplo já
apresenta a
vantagem de utilizar
a estrutura de vetor

Estrutura de Dados Homogêneas

Exemplo de Vetores

Este exemplo
apresenta o uso de
uma constante global

```
1  /*Calcula a média de cinco notas (usa vetor) */
2  #include<stdio.h>
3  #include<conio.h>
4  #define TAM 5
5
6  int main(){
7      float notas[TAM], media=0.0;
8      int i;
9
10     for(i=0; i<TAM; i++){
11         printf("\n Digite a nota do aluno %d", i+1);
12         scanf("%f", &notas[i]);
13         media += notas[i];
14     }
15     media /= 5.0;
16     printf("\n Média das Notas: %.2f ", media);
17
18     getch();
19     return 0;
20 }
```

Estrutura de Dados Homogêneas

Exercícios de Vetores

- 1 – Crie um programa que leia do teclado seis valores inteiros e armazene em um vetor, em seguida mostre na tela os valores lidos na ordem inversa.
- 2 – Crie um programa que leia um vetor de 10 posições. Escreva na tela quantos valores pares foram armazenados nesse vetor.
- 3 – Faça um programa que preencha um vetor com 10 elementos inteiros. Calcule e mostre apenas os números ímpares e qual a quantidade de números ímpares.
- 4 – Faça um programa que receba do usuário um vetor X com 10 posições. Em seguida, deverão ser impressos o maior e o menor elemento desse vetor.

Estrutura de Dados Homogêneas

Exercícios de Vetores

- 5 – Crie um programa que leia um vetor com 8 posições. Em seguida mostre na tela somente os valores que se encontram nas posições pares do vetor (considere a posição zero).
- 6 – Crie um programa que leia um vetor de 10 posições. Escreva na tela quais são as posições do vetor que se encontram com valores ímpares.
- 7 – Faça um programa que preencha um vetor com 10 elementos inteiros. Calcule e mostre a média dos valores do vetor.
- 8 – Faça um programa que leia um vetor VET de 10 posições. Mostre somente os números divisíveis por 2, a sua quantidade e quantos não são divisíveis por 2.



ESTRUTURA DE DADOS E ALGORITMOS I

Estruturas Homogêneas -

Vetores

Profº. Sérgio Roberto Costa Vieira, M.Sc.

Cursos de Computação

2º. Período