

Vetores

Um vetor é um tipo de variável capaz de armazenar uma coleção de valores de um mesmo tipo de dados.

Cada um dos valores armazenados num vetor é denominado *item*.

O item é identificado por um número natural, a partir do zero, denominado *índice*.

Para indicar que uma variável é um vetor, devemos declará-la com o sufixo $[n]$, na qual n é uma constante que estabelece o tamanho do vetor.

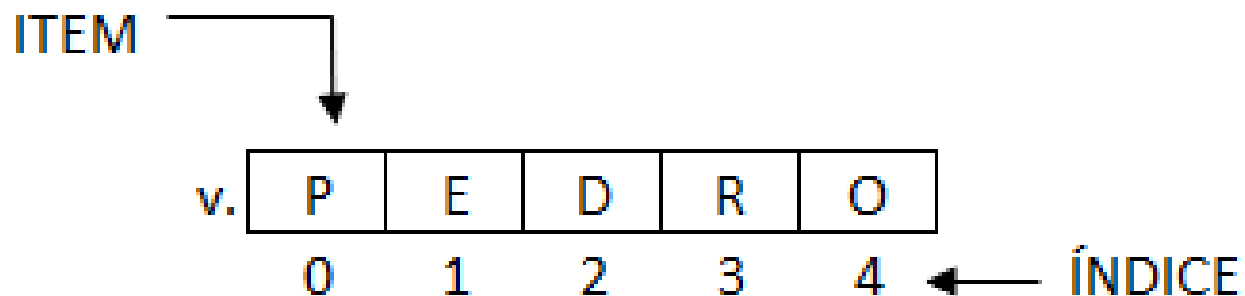
Vetores

Por exemplo, a declaração “char v[5]” cria um vetor com 5 posições, cada qual capaz de armazenar um caractere.

Além disso, com esta única declaração, criamos as variáveis:

$v[0]$, $v[1]$, $v[2]$, $v[3]$, $v[4]$

Nota que, como a indexação se inicia em 0, o último item de um vetor de tamanho n é armazenado na posição $[n - 1]$.



Vetores

Exemplo 1:

Faça um programa em C que leia 5 números e, em seguida, exiba todos os números lidos na tela.

```
#include <stdio.h>
#include <stdlib.h> /* funções envolvendo alocação de memória, controle de processos, conversões e outras.*/

int main(int argc, char* argv[])
{
    int V[5], i;
    for (i=0; i<=4; i++) {
        printf("Digite V[%d] = ", i);
        scanf("%d", &V[i]);
    }
    printf("\n\nOs valores lidos sao: \n");
    for (i=0; i<=4; i++) {
        printf("%d\t", V[i]);
    }
    return 0;
}
```

```
Digite V[0] = 1
Digite V[1] = 2
Digite V[2] = 3
Digite V[3] = 4
Digite V[4] = 5
```

Os valores lidos sao:

```
1      2      3      4      5
-----
```

```
Process exited after 14 seconds with return value 0
Pressione qualquer tecla para continuar. . .
```

Vetores

Exemplo 2:

Faça um programa em C que leia 5 números e, em seguida, exiba todos os números lidos na tela em ordem inversa.

```
#include <stdio.h>
#include <stdlib.h> /* funções envolvendo alocação de memória, controle de processos, conversões e outras.*/

int main(int argc, char* argv[])
{
    int V[5], i;
    for (i=0; i<=4; i++) {
        printf("Digite V[%d] = ", i);
        scanf("%d", &V[i]);
    }
    printf("\n\nOs valores lidos sao: \n");
    for (i=4; i>=0; i--) {
        printf("%d\t", V[i]);
    }
    return 0;
}
```

```
Digite V[0] = 1
Digite V[1] = 2
Digite V[2] = 3
Digite V[3] = 4
Digite V[4] = 5
```

```
Os valores lidos sao:
```

```
5         4         3         2         1
```

```
-----
```

```
Process exited after 13.73 seconds with return value 0
Pressione qualquer tecla para continuar. . .
```

Vetores

Exemplo 3:

Dadas as temperaturas registadas diariamente, durante uma semana, escreva um programa em C que informe em quantos dias a temperatura ficou acima da média.

```
#include <stdio.h>
#include <stdlib.h> /* funções envolvendo alocação de memória, controle de processos, conversões e outras.*/

int main(int argc, char* argv[])
{
    int i, qtd;
    float V[7], soma, media;
    for (i=0; i<=6; i++)
    {
        printf("Digite a %da. temperatura = ",i+1);
        scanf("%f",&V[i]);
        soma = soma + V[i];
    }
    media = soma / 7;
    for (i=0; i<=6; i++) if (V[i] > media) qtd++;
    printf("A media foi %.2f\n",media);
    printf("A temperatura ficou acima da media: %d dias",qtd-1);
    return 0;
}
```

```
Digite a 1a. temperatura = 1
Digite a 2a. temperatura = 2
Digite a 3a. temperatura = 3
Digite a 4a. temperatura = 4
Digite a 5a. temperatura = 5
Digite a 6a. temperatura = 6
Digite a 7a. temperatura = 7
A media foi 4.00
A temperatura ficou acima da media: 3 dias
-----
Process exited after 11.5 seconds with return value 0
Pressione qualquer tecla para continuar. . .
```

Vetores

Exemplo 4:

Faça um programa em C que leia 5 números e, em seguida, exiba na tela o maior valor lido e a posição em que ele se encontra.

```
#include <stdio.h>

int main(int argc, char* argv[])
{
    int V_lido[5], i, maior, posicao=0;
    for (i=0; i<=4; i++) {
        printf("Digite V[%d] = ", i);
        scanf("%d", &V_lido[i]);
    }
    maior = V_lido[0];
    for (i=0; i<=4; i++){
        if (V_lido[i] > maior){
            maior = V_lido[i];
            posicao = i;
        }
    }
    printf("\n\nMaior valor: %d\n", maior);
    printf("\n\nPosicao: %d\n", posicao);
    return 0;
}
```

Vetores

Exemplo 5:

Faça um programa em C que leia 5 números e, em seguida, exiba todos os números lidos na tela em ordem crescente.

```
#include <stdio.h>

int main(int argc, char* argv[])
{
    int vetor[5], i, j, aux;
    for (i=0; i<=4; i++) {
        printf("Digite V[%d] = ", i);
        scanf("%d", &vetor[i]);
    }
    for (i = 0; i <=4; i++) {
        for (j = 0; j <=4; j++) {
            if (vetor[i] < vetor[j]) {
                aux = vetor[i];
                vetor[i] = vetor[j];
                vetor[j] = aux;
            }
        }
    }
    for (i=0; i<=4; i++) {
        printf("%d\t", vetor[i]);
    }
    return 0;
}
```


Exercícios para assimilação

Atividade prática para treinar em casa

Prática para treinar em casa

Prática 8.1:

Elaborar uma aplicação console para ler 12 elementos de um vetor A. Construir um vetor B do mesmo tipo, observando a seguinte lei de formação: *“Todo elemento de B deverá ser o quadrado do elemento de A correspondente”*. Apresentar o vetor B.

Prática 8.2:

Elaborar uma aplicação console que leia 5 elementos de um vetor A. No final, apresentar o total da soma de todos os elementos que sejam ímpares.

Prática 8.3:

Elaborar uma aplicação console que preencha dois vetores com 5 elementos numéricos cada e depois ordene-os de maneira crescente. Deverá ser gerado um terceiro vetor com dez posições, composto pela junção dos elementos dos vetores anteriores, também ordenado de maneira crescente.



Obrigada!!!