



VETORES

Ana Cristina Ramos Alves

Programação de Computadores Aplicada a Engenharia de Agrimensura

O QUE SÃO VETORES?

- Um vetor é uma sequência de vários valores do mesmo tipo, armazenados sequencialmente na memória, e fazendo uso de um mesmo nome de variável para acessar esses valores.
- Na linguagem C, quando declaramos um vetor (conceito análogo ao de declaração de uma variável simples) devemos informar a dimensão do vetor, isto é, o número máximo de elementos que poderá ser armazenado no espaço de memória que é reservado para o vetor.
- Devemos também informar o tipo dos valores que serão armazenados no vetor (por exemplo, int, float, char ou double).
- Em um vetor, só podemos armazenar valores de um mesmo tipo.

COMO DECLARAR UM VETOR?

Um vetor deve ser declarado da seguinte forma:

Os elementos do vetor são indexados de 0 até $n-1$, onde n é o tamanho do vetor.

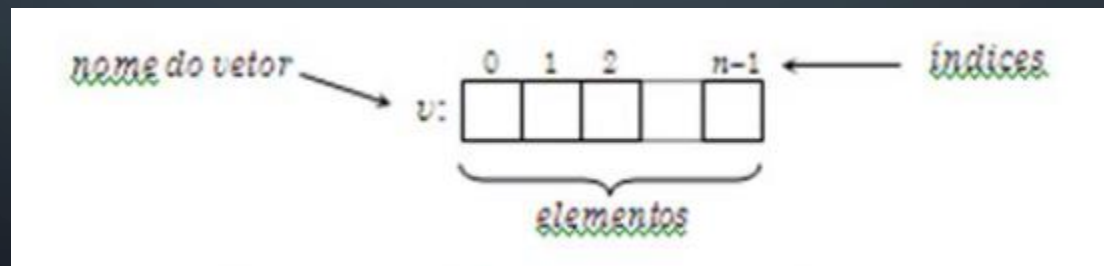
```
tipo nome [ tamanho ];
```

- Exemplos:

```
int idades[30];
```

```
double alturas[30];
```

Os elementos são acessados com o operador [], onde o primeiro elemento está na posição 0, o segundo na posição 1, assim sucessivamente até a posição tamanho-1.



EXEMPLOS

1) Leia 10 números inteiros positivos e armazene em um vetor. Em seguida determine e imprima o maior elemento no vetor.

```
#include<stdio.h>
int main()
{
    int v[10], i, maior;                                /* O vetor possui 10 posições */
    maior = 0;
    for (i=0; i<10; i++){
        printf("Digite um valor: ");
        scanf("%d", &v[i]);
    }
    for (i=0; i<10; i++){
        if (v[i] > maior)                                /* Se o vetor na posição i for maior que o maior valor */
            maior = v[i];                                /* O maior valor receberá tal valor armazenado na posição i */
    }
    printf("O maior elemento do vetor é = %d\n", maior);
    return 0;
}
```

2) Deseja-se publicar o número de acertos de cada aluno em uma prova em forma de testes. A prova consta de 30 questões, cada uma com cinco alternativas identificadas por A, B, C, D e E. Para isso são dados:

- o cartão gabarito;
- o número de alunos da turma;
- o cartão de respostas para cada aluno, contendo o seu número e suas respostas.

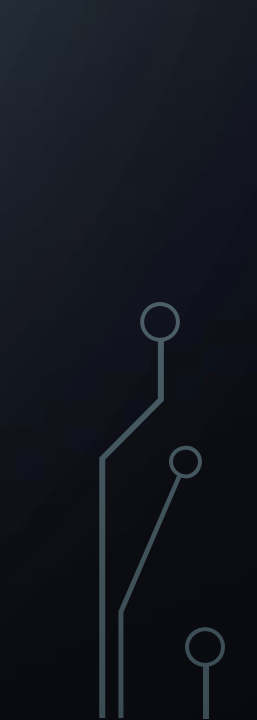

```
#include <stdio.h>
```

```
int main() {  
    int n, i, aluno;  
    char gabarito[30], resp[30];  
    printf("Digite o numero de alunos: ");  
    scanf("%d", &n);  
    printf("Digite o gabarito da prova: ");  
    for (i = 0; i < 30; i++)  
        scanf(" %c", &gabarito[i]);  
    for (aluno = 1; aluno <= n; aluno++)  
    {  
        pontos = 0;  
        printf("Digite as respostas do aluno %d: ", aluno);  
        for (i = 0; i < 30; i++) {  
            scanf(" %c", &resp[i]); /* atencao para o espaco em branco */  
            if (resp[i] == gabarito[i])  
                pontos = pontos + 1;  
        }  
        printf("O aluno %d fez %d pontos\n", aluno, pontos);  
    }  
    return 0;  
}
```



Exercícios do livro: Fundamentos para Programação de Computadores – Ana Ascencio

Vamos agora resolver alguns exercícios para fixar o conteúdo aprendido!



(Exercício 1, cap.6 – pág. 159) - Faça um programa que preencha um vetor com nove números inteiros, calcule e mostre os números primos e suas respectivas posições.

```
#include <stdio.h>
int main()
{
    int num[9], i, j, cont;    /*declarar o tipo e tamanho do vetor*/
    for (i=0;i<9;i++)
    {
        printf("Digite o valor: ", i++); /*deverá informar nove valores*/
        scanf("%d", &num[i]); /*cada valor será armazenado em uma posição na memória*/
    }
    for (i=0;i<9;i++){
        cont = 0;
        for (j=1;j<=num[i];j++){
            if ((num[i]%j) == 0) /*se o resto do valor digitado for igual a zero, então o valor não é primo*/
                cont++;
        }
        if (cont == 2) /*números primos são divididos por 1 e por ele mesmo. Para ser primo, o resto deve ser menor ou igual a
2*/
            printf("\nO número %d é primo e ocupa a posição %d");
    }
    return 0;
}
```

(Exercício 18, cap.6 - pág. 180-181) Faça um programa que receba o número sorteado por um dado em vinte jogadas. Mostre os números sorteados e a frequência com que apareceram.

```
#include <stdio.h>
int main()
{
    int dado[20], i, num1, num2, num3, num4, num5, num6; /*o vetor dado receberá vinte jogadas*/
    for (i=0;i<20;i++){
        printf("\nDigite um valor de um dado entre 1 e 6: ");
        scanf("%d", &dado[i]);
        while ((dado[i]<1) || (dado[i]>6)){ /*enquanto o valor do dado for menor que 1 ou maior que 6, será inválido*/
            printf("\nNúmero inválido, digite novamente: ");
            scanf("%d", &dado[i]);
        }
    }
    printf("\n\nLista dos numeros sorteados ");
    for (i=0;i<20;i++) /*para as vinte jogadas, o dado apresenta 6 lados diferentes*/
        scanf("%d", &dado[i]);
    num1 = 0;
    num2 = 0;
    num3 = 0;
    num4 = 0;
    num5 = 0;
    num6 = 0;
```


continuação....

```
for (i=0;i<20;i++){ /*se o valor armazenado na jogada for igual ao valor do dado */
    if (dado[i] == 1)
        num1++;
    if (dado[i] == 2)
        num2++;
    if (dado[i] == 3)
        num3++;
    if (dado[i] == 4)
        num4++;
    if (dado[i] == 5)
        num5++;
    if (dado[i] == 6)
        num6++;
    } /*então imprime o numero sorteado e a quantidade de vezes que foi sorteado*/
    printf("\nO numero 1 foi sorteado %d vez(es)", num1);
    printf("\nO numero 2 foi sorteado %d vez(es)", num2);
    printf("\nO numero 3 foi sorteado %d vez(es)", num3);
    printf("\nO numero 4 foi sorteado %d vez(es)", num4);
    printf("\nO numero 5 foi sorteado %d vez(es)", num5);
    printf("\nO numero 6 foi sorteado %d vez(es)", num6);
    return 0;
}
```

The background is a dark blue gradient. In the corners, there are white line-art illustrations of circuit boards or neural networks, with lines and small circles representing components.

Obrigada pela atenção!