

RESUMO DE VETOR E MATRIZ

VETOR:

O vetor pode ser chamado de array também, ele consiste em armazenar variáveis do mesmo tipo sob um único identificador ou nome.

Como por exemplo:

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

```
int main() {  
  
    int vetor[5] = {1, 2, 3, 4, 5};  
  
    for (int i = 0; i < 5; i++) {  
  
        printf("Elemento %d: %d\n", i, vetor[i]);  
  
    }  
  
    return 0;  
  
}
```

Nesse exemplo, o vetor já está com seus valores definidos.

A saída desse pequeno código ficará da seguinte forma:

Elemento 0: 1

Elemento 1: 2

Elemento 2: 3

Elemento 3: 4

Elemento 4: 5

No entanto, para entender como especificar o tamanho do vetor, ou seja, quantas informações irá reter. Precisa entender o que já está incluso como informação.

O vetor sempre começa na casa 0, o último número do vetor pertence ao caractere nulo '\0' (isso para caracteres do tipo char). Por tanto, quando um vetor não é do tipo char, no momento que for declarado `v[5]`, se for de um em um a saída seria 0,1,2,3,4. Veja que contém 5 números, ou seja, os índices de um vetor em C são baseados em zero, e o último índice é sempre o tamanho do vetor menos um.

É importante sempre lembrar que os números que for armazenado no vetor seja do mesmo tipo de variável.

Em suma, vale lembrar que os vetores em C começam seus índices a partir de 0. A declaração `v[5]` significa que há cinco elementos no vetor, com índices 0, 1, 2, 3, e 4. Isso é fundamental para acessar os elementos corretamente. Além disso, ao lidar com vetores de caracteres (strings), o caractere nulo '\0' é crucial para indicar o final da string. No entanto, é importante notar que essa consideração específica do caractere nulo é mais relevante para strings do que para vetores de outros tipos. O uso de um loop `for` é comum para percorrer os elementos de um vetor de maneira sistemática, como exemplificado no código acima.

MATRIZ:

Uma matriz na programação é como se fosse um vetor dentro de um vetor.

É uma estrutura bidimensional que armazena elementos em linhas e colunas. Cada elemento da matriz é representado por um par de índice, o primeiro do par sendo a linha e o segundo sendo a coluna.

Um exemplo para melhor compreensão:

Como no exemplo do vetor, será dado os dados da matriz.

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

```

int main() {

    // Declarando e inicializando uma matriz 3x3

    int matriz[3][3] = {

        { 1, 2, 3 },

        { 4, 5, 6 },

        { 7, 8, 9 }

    };


    // Acessando e exibindo os elementos da matriz

    // o i pertence para os valores das linhas

    for (int i = 0; i < 3; i++) {

        // o j é os valores para as colunas

        for (int j = 0; j < 3; j++) {

            printf("Elemento [%d][%d]: %d\n", i, j, matriz[i][j]);

        }

    }


    return 0;

}

```

Ou seja, o código para a matriz anda em linha, dessa forma adquiri o todos os valores da linha 0 (já que inicia em 0), e fica no laço do segundo for obtendo todos os valores das colunas. Esse processo é repetido até que consiga obter ou nesse caso relacionar todos os números da matriz já fornecida.

Observe a saída do código para melhor entendimento:

Elemento [0][0]: 1

Elemento [0][1]: 2

Elemento [0][2]: 3

Elemento [1][0]: 4

Elemento [1][1]: 5

Elemento [1][2]: 6

Elemento [2][0]: 7

Elemento [2][1]: 8

Elemento [2][2]: 9

Curiosidade: as matrizes podem ser tridimensionais ou mais complexas do que essas, porém é mais usada a bidimensional.