

LP – Linguagem de Programação I

MATRIZES - Revisão

Matriz é uma coleção de variáveis do mesmo tipo que é referenciada por um nome comum.

Um elemento específico em uma matriz é acessado por meio de um índice. Em C, todas as matrizes consistem em posições contíguas na memória. O endereço mais baixo corresponde ao primeiro elemento e o mais alto, ao último.

Matrizes podem ter várias dimensões.

Unidimensional:

```
double balance[100]
```

A quantidade de armazenamento necessário para guardar uma matriz está diretamente relacionada com seu tamanho e seu tipo. Para uma matriz unidimensional, o tamanho total em bytes é calculado da seguinte forma:

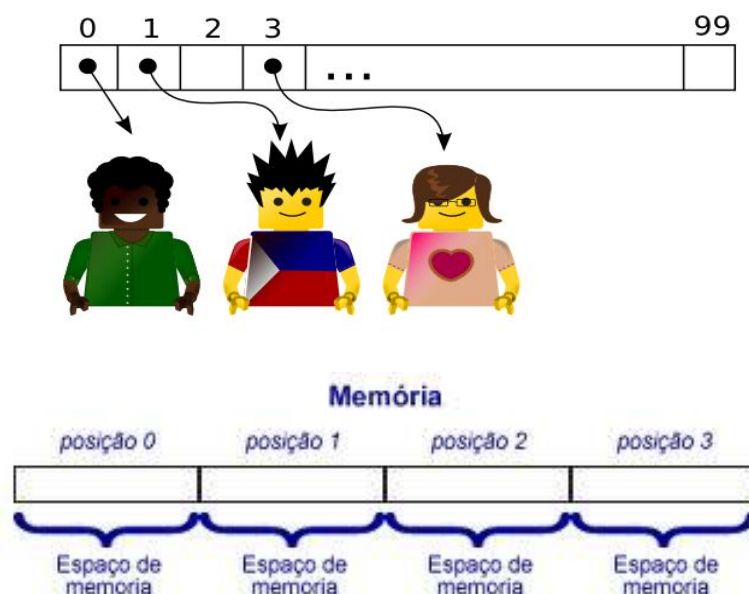
Total em bytes = sizeof(tipo) * tamanho da matriz

C não tem verificação de limites em matrizes. Você poderia ultrapassar o fim de uma matriz e escrever nos dados de alguma outra variável, por exemplo. Como programador, é seu trabalho prover verificação dos limites onde necessário.

```
Ex.: int numero[10], i;
```

```
for (i=0;i<100;i++) numero[i]=i;
```

Resumindo, matrizes unidimensionais são, essencialmente, listas de informações do mesmo tipo, que são armazenadas em posições contíguas da memória em uma ordem de índice.



Bidimensional:

C suporta matrizes multidimensionais. A forma mais simples de matriz multidimensional é a matriz bidimensional – uma matriz de matrizes unidimensionais. Declaração:

```
int matriz[10][20]
```

Exemplo para carregar uma matriz 3x4 chamada num com números de 1 a 12.

num[0][0] tem o valor 1, num[0][1], o valor 2 e assim por diante.

	0	1	2	3
0	1	2	3	4
1	5	6	7	8
2	9	10	11	12

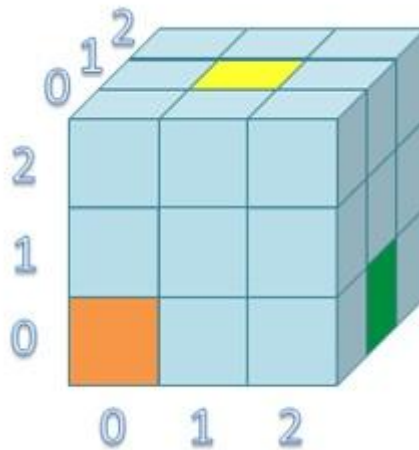
Matrizes bidimensionais são armazenadas em uma matriz linha-coluna, onde o primeiro índice indica a linha e o segundo indica a coluna.

No caso de uma matriz bidimensional, a seguinte fórmula fornece o número de bytes de memória necessários para armazená-la:

Bytes=tamanho do 1º índice * tamanho do 2º índice * sizeof(tipo base)

Portanto, assumindo inteiros de dois bytes, uma matriz de inteiros com dimensões 10,5 teria:

10*5*2 ou 100 bytes alocados.

Tridimensional:

A figura acima ilustra o que seria um array tridimensional. Não foram definidos valores para os elementos, apenas cores diferentes a alguns itens para que se possa referenciá-los. Neste caso, cada elemento é identificado por três índices. Na figura, o elemento de índices (0,0,0) é o de cor laranja, enquanto o elemento (0, 2, 1) é o verde, o elemento amarelo (no topo), possui índices (2, 1, 1).