

UNIVALI CTTMar Kobrasol - São José - Ciência da Computação 1per
VARIÁVEL HOMOGÊNEA MULTIDIMENSIONAL – MATRIZ

Definição: corresponde a posições consecutivas de memória, identificadas por um mesmo nome, **individualizadas por 2 índices (linha e coluna)** e cujo conteúdo é do mesmo tipo.

⇒ cada elemento da matriz é tratado como se fosse uma variável simples. Para referenciar um elemento da matriz utiliza-se o nome dela e a identificação do elemento (linha e coluna) entre colchetes.

⇒ Muito semelhante ao vetor.

Especificação de matriz (bidimensional) em algoritmo:

nome_matriz: **VETOR** [linhaInicial .. linhaFinal, colunaInicial .. colunaFinal] de tipo

Ex.: var

Tabela: vetor [1..3,1..4] de inteiro

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

NA LINGUAGEM C++

Os elementos da matriz (como no vetor) são sempre numerados por índices INTEIROS iniciados em 0 (zero).

- declaração:

tipo nome_matriz [num_linhas, num_colunas];

Ex.: unsigned int votos[3][5];

		regiões				
candidatos		0	1	2	3	4
	0	100000	2000	...		
	1	...				
	2					150000

UNIVALI CTTMar Kobrasol - São José - Ciência da Computação 1per
VARIÁVEL HOMOGÊNEA MULTIDIMENSIONAL – MATRIZ

EXEMPLO – VISUALG:

1. Algoritmo "matriz"
2. var
3. matA: vetor [1..3,1..3] de inteiro { matriz 3x3 }
4. lin, col: inteiro
5. inicio
6. escreval ("Entrada")
7. para lin de 1 ate 3 faca
8. para col de 1 ate 3 faca
9. escreva ("Elemento [", lin, ",", col, "]: ")
10. leia (matA[lin,col])
11. fimpara
12. fimpara
13. escreval ("Saida")
14. para lin de 1 ate 3 faca
15. para col de 1 ate 3 faca
16. escreva (matA[lin,col], " ")
17. fimpara
18. escreval {pula linha}
19. fimpara
20. finalgoritmo

EXEMPLO – C++:

```
#include <iomanip>
using namespace std;
int main()
{
    int lin, col, matA[3][3]; // matriz 3x3
    cout << "Entrada" << endl;
    for(lin=0; lin < 3; lin++)
        for(col=0; col < 3; col++) {
            cout << "Elemento [" << lin+1 << ", " << col+1 << "]: ";
            cin >> matA[lin][col];
        }
    cout << endl << "Saida" << endl;
    for(lin=0; lin < 3; lin++) {
        for(col=0; col < 3; col++) {
            cout << matA[lin][col] << " ";
        }
        cout << endl;
    }
    return 1;
}
```

UNIVALI CTTMar Kobrasol - São José - Ciência da Computação 1per
VARIÁVEL HOMOGÊNEA MULTIDIMENSIONAL – MATRIZ

EXEMPLO – PORTUGOL:

```
programa
{
    inclua biblioteca Util --> u
    funcao inicio()
    {
        const inteiro TAMMAX = 5 // Define a dimensão máxima da matriz
        inteiro matriz[TAMMAX][ TAMMAX] // Cria matriz c/tamanho máximo 5x5
        inteiro nlinhas, ncolunas // Dimensões reais da matriz

        escreva("Informe o numero de linhas da  matriz")
        faca
            leia(nlinhas)
        enquanto (nlinhas<=0 ou nlinhas>TAMMAX)

        escreva("Informe o numero de colunas da  matriz")
        faca
            leia(ncolunas)
        enquanto (ncolunas<=0 ou ncolunas>TAMMAX)

        para (inteiro linha = 0; linha < nlinhas; linha++)
        {
            para (inteiro coluna = 0; coluna < ncolunas; coluna++)
            {
                matriz[linha][coluna] = u.sorteia(1, 9) // Atribui valor aleatório
            }
        }
        para (inteiro linha = 0; linha < nlinhas; linha++)
        {
            para (inteiro coluna = 0; coluna < ncolunas; coluna++)
            {
                escreva("[", matriz[linha][coluna], "]") // Exibe o valor contido
            }
            escreva ("\n") // para pular linha
        }
    }
}
```

Obs.: usar o recurso do TAMMAX é interessante no que se refere a flexibilidade de manutenção (caso o tamanho máximo da matriz seja alterado para 100, basta alterar a instrução **const** inteiro TAMMAX = 5 para **const** inteiro TAMMAX = 100, e recompilar o programa).