

- ❑ Cada estrutura do tipo **vetor** armazena os dados numa **única** “fileira”;
- ❑ Há situações que a natureza dos dados nos indica que a sua **forma de armazenamento** possui mais de uma **dimensão**, e para isto podemos criar **matrizes**;
- ❑ Em programação uma **matriz** é um **vetor** que **possui mais de uma dimensão**;
- ❑ Sendo **bidimensional**, a variável atua como uma **grade de linhas e colunas**, onde o seu valor é a **interseção** entre um **linha e uma coluna**;

# Declaração

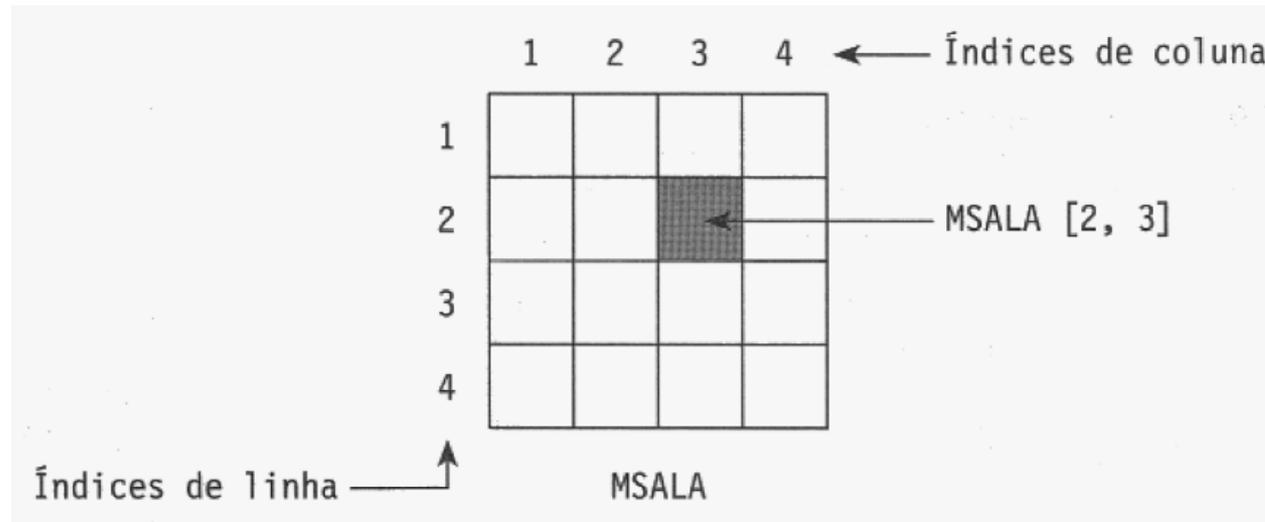
## Exemplos:

- a) tipo **M** = matriz [1..3, 2..4, 3..4] de reais;  
**Mat : M**
  
- b) tipo **SALA** = matriz [1..4, 1..4] de inteiros;  
**mSala: SALA;**

# Manipulação

- ❑ Para acessar um elemento em uma estrutura composta multidimensional – matriz – precisamos como em um edifício, de seu nome, de seu andar e de seu apartamento;
- ❑ Considerando um estrutura bidimensional (dois índices: andar e apartamento), o primeiro índice indica a linha e o segundo, a coluna;

Exemplo: mSala[2,3]



# Manipulação

## □ Atribuição:

**tipo M = matriz [1..3, 2..4, 3..4] de reais;**

**mat : M;**

**mat[1,1] <- 8.8;**

**mat[2,3] <- 7.3;**

# Exemplo

- Gerar e exibir a matriz abaixo:

```
11 12 13 14
15 16 17 18
19 20 21 22
```

```
program matriz
var
```

```
    tipo MATRIZ: matriz[1..3, 1..4] de inteiro;
    mat: MATRIZ;
    i, j, cont : inteiro;
```

```
inicio
```

```
    para i de 1 ate 3 faca
        para j de 1 ate 4 faca
            mat [i,j] <- cont;
            cont <- cont + 1;
```

```
        fimpara;
```

```
    fimpara;
```

```
    para i de 1 ate 3 faca
        escreva( mat[i, 1], " ", mat[i,2], " ", mat[i,3], " ", mat[i,4]);
```

```
    fimpara;
```

```
fim.
```

# Exemplo

- **Construir um algoritmo que efetue, a soma e a impressão do resultado entre duas matrizes inteiras que comportem 25 elementos;**

```
program matriz
var
    tipo MATRIZ: matriz[1..5, 1..5] de inteiro;
    ma, mb, mr: MATRIZ;
    i, j : inteiro;
inicio
    i <- 1;

    enquanto ( i <= 5 ) faca
        j <- 1;
        enquanto ( j <= 5 ) faca
            leia ( ma[ i, j ], mb [ i, j ] );
            mr[ i, j ] <- ma[ i, j ] + mb [ i, j ]
            j <- j + 1;
        fimenquanto;
        i <- i + 1;
    fimenquanto;

    j <- 1;
    enquanto ( j <= 5 ) faca
        i <- 1;
        enquanto ( i <= 5 ) faca
            escreva ( mr [ i, j ] );
            i <- i + 1;
        fimenquanto;
        j <- j + 1;
    fimenquanto;

fim.
```