

Algoritmos e Programação

Matrizes

Prof. Dr. Joaquim Assunção

CENTRO DE TECNOLOGIA
UFSM
2025



- ... Semelhante aos vetores, **porém** são estruturas **bidimensionais**.

Declaração - Exemplos

```
inteiro : minhaMat[5, 5];
```

```
inteiro : minhaMat[0..4, 0..4];
```

```
inteiro : minhaMat[0:4, 0:4];
```

```
inteiro : minhaMat[ , ];
```

```
minhaMat : vetor[0..4, 0..4] de inteiro;
```

```
minhaMat : vetor[0:4, 0:4] de inteiro;
```

```
//Todas estas também podem ser feitas com [][] no lugar de  
[,]
```

Descrição

- Considere uma matriz um vetor de duas dimensões.

```
inteiro: minhaMat[3][3];
```

```
minhaMat[2,1] ← 1;
```

```
minhaMat[1,2] ← 2;
```

| | | |
|--|---|---|
| | | |
| | | 2 |
| | 1 | |

Descrição

- Considere uma matriz um vetor de duas dimensões.

```
inteiro : minhaMat[3][3];
```

Com estruturas de repetição, podemos preencher da seguinte maneira:

```
PARA i=0 ATE 2 FACA
```

```
    PARA j=0 ATE 2 FACA
```

```
        minhaMat[i,j] ← i+1
```

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 2 | 2 | 2 |
| 3 | 3 | 3 |

Descrição

- Considere uma matriz um vetor de duas dimensões.

```
inteiro minhaMat[3][3];
```

Com estruturas de repetição, podemos pesquisar da seguinte maneira:

```
PARA i=0 ATE 2 FACA
```

```
    PARA j=0 ATE 2 FACA
```

```
        SE (minhaMat[i,j] == 2)
```

```
            print("segunda linha");
```

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 2 | 2 | 2 |
| 3 | 3 | 3 |

Demonstração

- Crie a seguinte matriz →

Após, imprima seu elemento central.

| | | |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

```
//pseudo código
int i, j, mat[3][3], cont = 0;

para (i de 0 ate 2){
    para (j de 0 ate 2){
        cont += 1;
        mat[i][j] = cont;
    }
}
escreva(mat[1][1]);
}
```

```
//C++
int main() {
    int i, j, mat[3][3], cont = 0;

    for (i = 0; i < 3; i++){
        for (j = 0; j < 3; j++){
            cont += 1;
            mat[i][j] = cont;
        }
    }
    cout << mat[1][1];
}
```

Exercícios

1. Crie um algoritmo que receba uma matriz e imprima os números de sua diagonal principal caso a soma desses números seja maior do que 10.
2. Crie um algoritmo que receba uma matriz e imprima os números de sua diagonal principal caso não haja números menores do que 9 na diagonal principal.

Exercício 1

```
// → Complete para fazer o Código compilar.  
// soma os elementos da diagonal principal  
for (int i = 0; i < N; i++) {  
    soma += mat[i][i];  
}  
  
if (soma > 10) {  
    cout << "Elementos da diagonal principal:" << endl;  
    for (int i = 0; i < N; i++) {  
        cout << mat[i][i] << endl;  
    }  
} else {  
    cout << "A soma da diagonal (" << soma << ") nao é  
maior que 10." << endl;  
}
```

Exercício

3. Crie um algoritmo que recebe duas matrizes quadradas, de mesmo tamanho, uma de inteiros e outra de caracteres, e retorna um vetor contendo a concatenação das duas, elemento por elemento.

Exemplo:

| | |
|---|---|
| 2 | 3 |
| 4 | 5 |

| | |
|---|---|
| a | B |
| c | d |

| | | | |
|----|----|----|----|
| 2a | 3B | 4c | 5d |
|----|----|----|----|