

Matrizes

Introdução à Ciência da Computação

Rosane Minghim

Apoio na confecção: Carlos Elias A. Zampieri

Danilo Medeiros Eler

Rogério Eduardo Garcia

Renato Rodrigues

Recordando: Vetores

- Tipos Compostos
 - Conjunto de vários elementos de mesmo tipo
- Um único nome. Ex: V
- Um tipo. Ex: $V:\text{inteiro}$



- Armazenamento em sequência. $V:\text{inteiro}[1..7]$
- Um índice.
 - Acesso direto: $V[i]$

Matrizes

- Tipos Compostos
 - Conjunto de vários elementos de mesmo tipo
- Um único nome. Ex: M
- Um tipo. Ex: M:inteiro

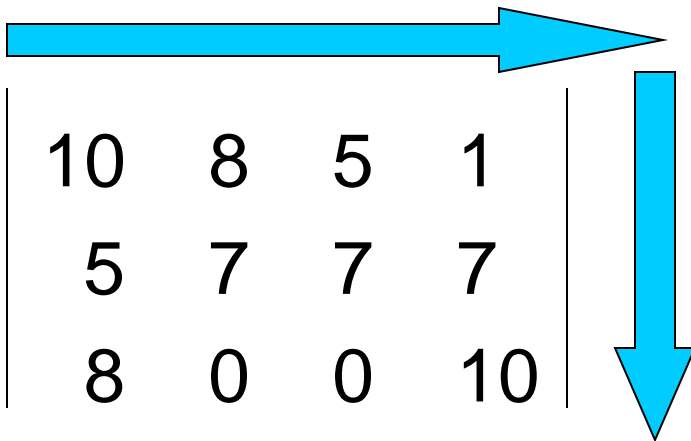
$$M = \begin{vmatrix} 10 & 8 & 5 & 1 \\ 5 & 7 & 7 & 7 \\ 8 & 0 & 0 & 10 \end{vmatrix}$$

Matrizes

- A indexação é similar à dos vetores
- Armazenamento em sequência para cada dimensão. $M:\text{inteiro}[1..3][1..4]$

M =

10	8	5	1
5	7	7	7
8	0	0	10



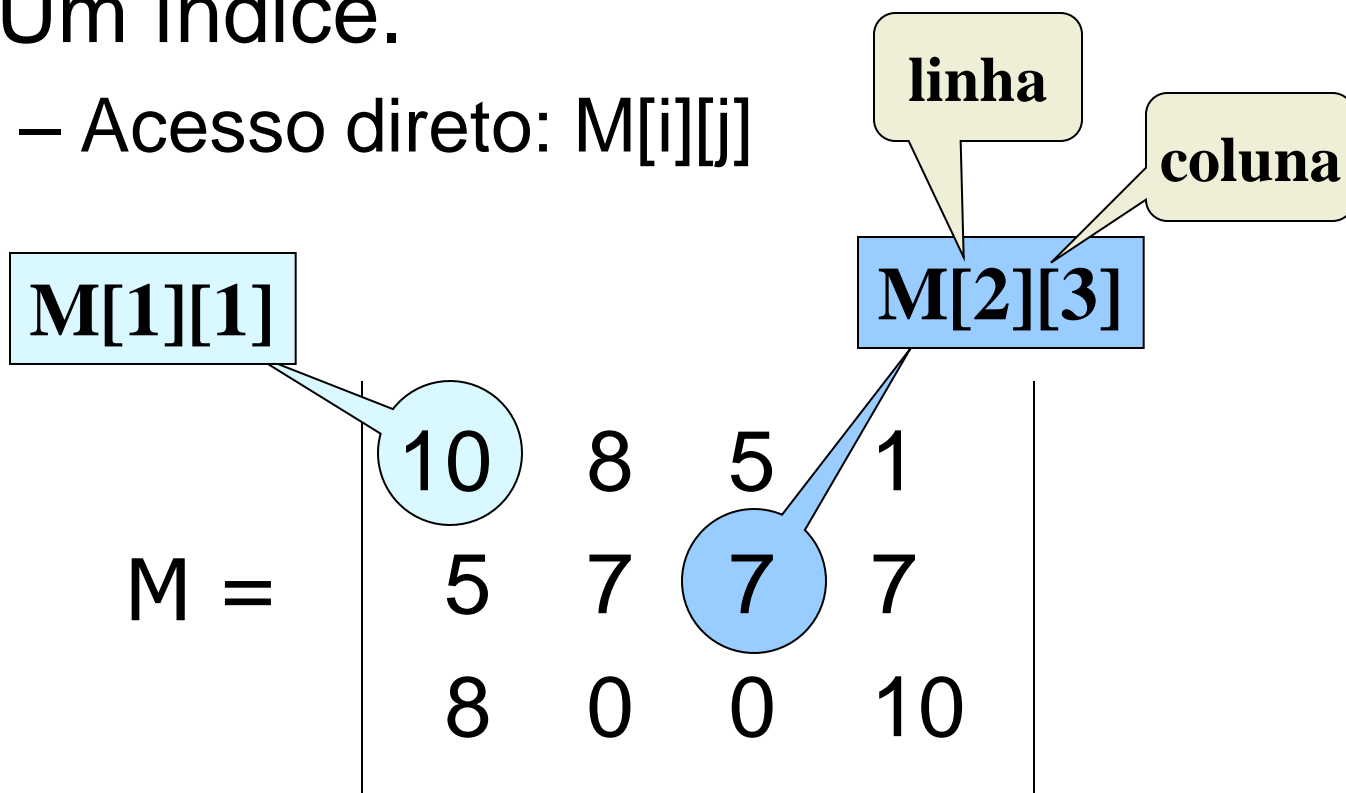
Genericamente, para n dimensões:

$M:\text{inteiro}[l_{i1},l_{s1}] [l_{i2},l_{s2}] \dots [l_{in},l_{sn}]$

Matrizes

- Um índice.

- Acesso direto: $M[i][j]$





Exemplos

```
tipo  matriz = real[1..10][1..5]
```

```
variável
```

```
    notas: matriz
```

```
notas[1][3] ← 7,5
```

No exemplo a matriz possui duas dimensões,
uma variando de 1 a 10 e a segunda de 1 a 5

Exemplo: média aritmética

Faça um algoritmo para ler as 5 notas dos alunos de uma turma, armazená-las numa matriz, e calcular suas médias finais.

```
Algoritmo média_provas
```

```
tipo matriz =  
    real[1..50][1..5]
```

```
variável
```

```
    soma, media: real  
    notas: matriz  
    lin, col, nro_alunos:  
    inteiro
```

```
leia(nro_alunos)
```

```
para lin de 1 até  
    nro_alunos faça  
        para col de 1 até 5 faça  
            leia(notas[lin][col])  
        fim para  
    fim para
```

```
para lin de 1 até  
    nro_alunos faça  
        soma ← 0  
        para col de 1 até 5 faça  
            soma ←  
            notas[lin][col] +soma  
        fim para  
        media ← soma / 5  
        escreva('aluno', lin, '  
            = ',media)  
    fim para  
fim
```

Exemplo: média aritmética

```
Algoritmo média_provas
tipo matriz =
  real[1..50][1..5]
variável
  soma, media: real
  notas: matriz
  lin, col, nro_alunos:
    inteiro

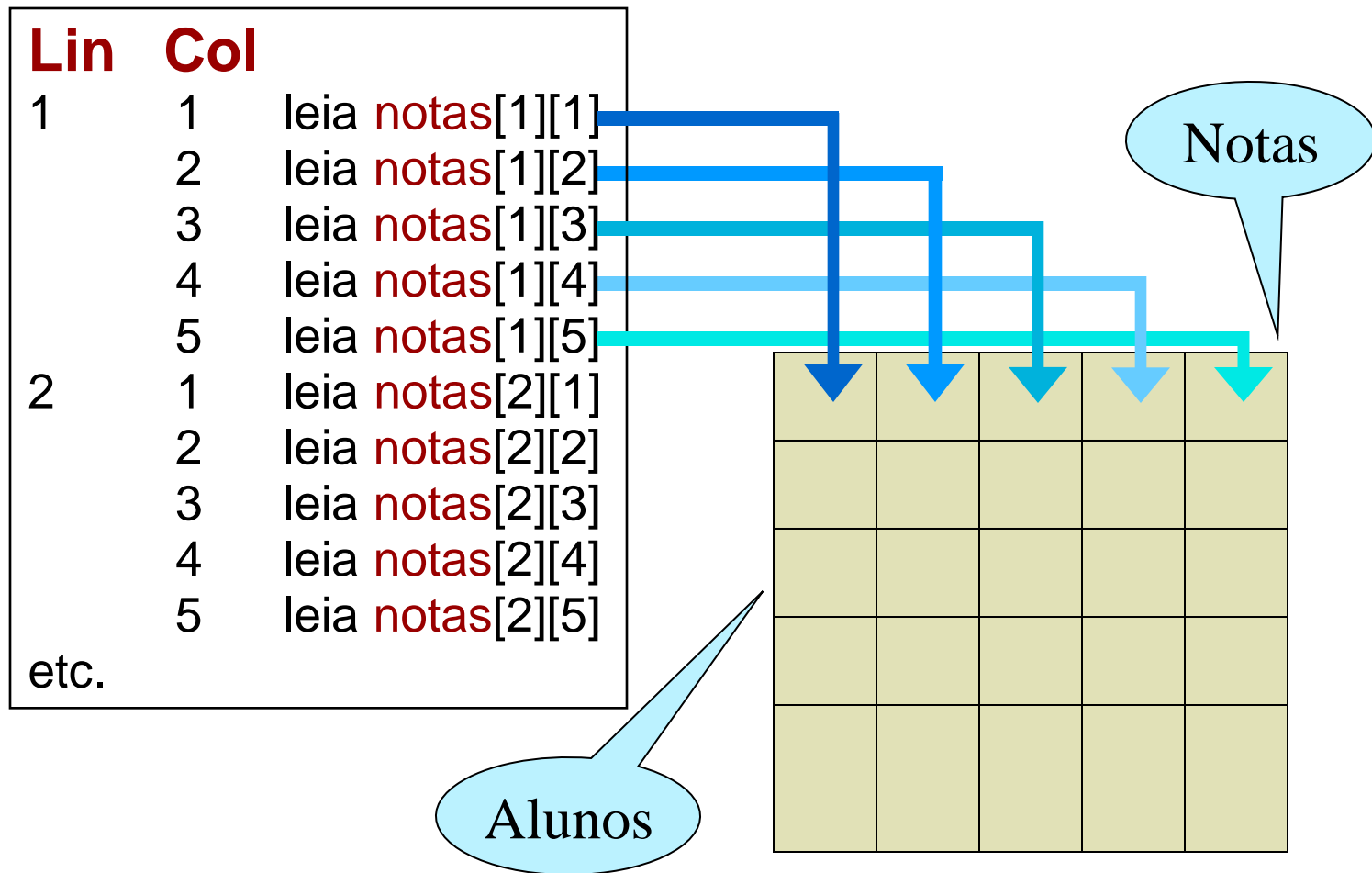
  leia(nro_alunos)
  para lin de 1 até
    nro_alunos faça
    para col de 1 até 5 faça
      leia(notas[lin][col])
    fim para
  fim para
```

Alunos

Notas

Lin	Col	
1	1	leia notas[1][1]
	2	leia notas[1][2]
	3	leia notas[1][3]
	4	leia notas[1][4]
	5	leia notas[1][5]
2	1	leia notas[2][1]
	2	leia notas[2][2]
	3	leia notas[2][3]
	4	leia notas[2][4]
	5	leia notas[2][5]
etc.		

Exemplo: média aritmética





Matrizes em Subprogramas

- Similar ao uso de vetores
- É importante para o uso de matrizes em subprogramas que o seu tipo seja definido previamente, por uma questão de consistência, compatibilidade de tipos e organização do código



Declaração de Tipos

```
mat_notas = real[1..50][1..5]  
objeto_logico = lógico[-10..10][-10..10][-10..10]
```

- Desta forma, subprogramas e algoritmos principais podem trabalhar sobre matrizes de mesmo tipo pré-definido
- Utilizando as definições acima, as declarações ficariam:

Variável

```
nota: mat_notas  
objeto: objeto_logico
```

Matrizes em subprogramas - Exemplo

O subprograma recebe a matriz e armazena os valores digitados.

```
Subprograma leia_matriz(mat, linhas, colunas)
```

```
e: linhas, colunas: inteiro {dimensões da matriz}
```

```
s: mat: matriz {valores lidos}
```

```
variável
```

```
    lin, col: inteiro
```

```
inicio
```

```
    para lin de 1 até linhas faça
```

```
        para col de 1 até colunas faça
```

```
            leia(mat[lin][col])
```

```
        fim para
```

```
    fim para
```

```
fim
```



Matrizes em subprogramas - Exemplo

O subprograma recebe a matriz e apresenta o conteúdo.

```
Subprograma escreva_matriz(mat, linhas,colunas)
```

```
e: linhas, colunas: inteiro {dimensões da matriz}  
   mat: matriz {notas}
```

```
variável
```

```
   lin, col: inteiro
```

```
inicio
```

```
   para lin de 1 até linhas faça
```

```
     para col de 1 até colunas faça
```

```
       escreva(mat[lin][col])
```

```
     fim para
```

```
   fim para
```

```
fim
```

Matrizes em subprogramas - Exemplo

O subprograma recebe a matriz e calcula as médias das linhas, armazenando o resultado num vetor.

```
Subprograma calcule_médias (mat, linhas, colunas, vet)
```

```
e: linhas, colunas: inteiro {dimensões da matriz}
```

```
e: mat: matriz {notas}
```

```
s: vet: vetor {medias}
```

```
variável
```

```
  lin, col: inteiro
```

```
inicio
```

```
  para lin de 1 até linhas faça
```

```
    vet[lin] ← 0
```

```
    para col de 1 até colunas faça
```

```
      vet[lin] ← vet[lin] + mat[lin][col]
```

```
    fim para
```

```
    vet[lin] ← vet[lin] / colunas
```

```
  fim para
```

```
fim
```



Matrizes em subprogramas - Exemplo

Algoritmo Principal:

Algoritmo média_provas

tipo

```
matriz = real[1..50][1..5]
```

```
vetor = real[1..50]
```

variável

```
notas: matriz
```

```
medias: vetor
```

```
nro_alunos: inteiro
```

```
leia(nro_alunos)
```

```
leia_matriz(notas,nro_alunos,5)
```

```
calcule_media(notas,nro_alunos,5, medias)
```

```
escreva_notas(notas,nro_alunos,5, medias)
```

```
fim para
```

Exercício: desenvolver o subprograma escreva_notas



Matrizes em subprogramas - Exemplo

O subprograma recebe a matriz apresenta o conteúdo.

```
Subprograma escreva_notas(mat, linhas,colunas,vet)
```

```
e: linhas, colunas: inteiro {dimensões da matriz}
   mat: matriz {notas}
   vet: vetor {medias}
```

```
variável
```

```
   lin, col: inteiro
```

```
inicio
```

```
   para lin de 1 até linhas faça
```

```
     para col de 1 até colunas faça
```

```
       escreva(mat[lin][col])
```

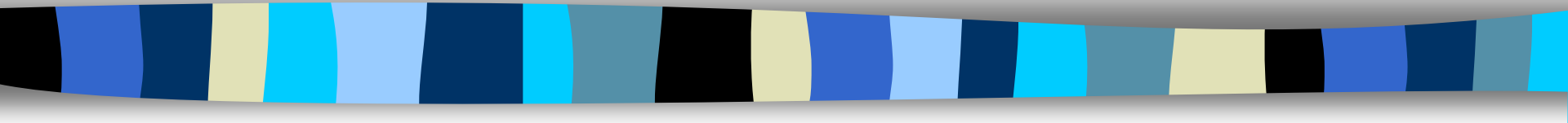
```
     fim para
```

```
   escreva (vet[lin])
```

```
   fim para
```

```
fim
```


Matrizes em C



Matrizes em C

- Linguagem C admite conjuntos de dados homogêneos com indexação em várias dimensões.
- A quantidade de posições em memória reservadas para a matriz é definida na declaração das dimensões pelo **número de linhas N** e o **colunas M**, ambos devem possuir valores inteiros. **Ex: int matriz[N][M];**
- Os formatos para declaração de variável e para definição de tipo com indexação multi-dimensional são apresentados a seguir:

```
int matriz[ 5 ][ 2 ];
```

```
float matriz[ 10 ][ 30 ];
```

ou

```
int matriz[ 3 ][ 3 ] = { 1,2,3, 4,5,6, 7,8,9 }; ou
```

```
int matriz[   ][ 3 ] = { 1,2,3, 4,5,6, 7,8,9 }; //somente a 1ª dimensão  
pode ser omitida
```

Matrizes em C

- Linguagem C admite conjuntos de dados homogêneos com indexação em várias dimensões.
- A quantidade de posições em memória reservadas para a matriz é definida na declaração das dimensões pelo **número de linhas** e **número de colunas**. Ex: int

Tipo dos elementos do conjunto

Nome da Variável

Número máximo de elementos nas linhas

Número máximo de elementos nas colunas

```
int matriz[ 5 ][ 2 ];  
float matriz[ 10 ][ 30 ];
```

ou

```
int matriz[ 3 ][ 3 ] = { 1,2,3, 4,5,6, 7,8,9 }; ou  
int matriz[   ][ 3 ] = { 1,2,3, 4,5,6, 7,8,9 }; //somente a 1ª dimensão  
pode ser omitida
```

Exemplo – Média Aritmética

```
#include <stdio.h>

#define NRO_NOTAS 5
#define NRO_ALUNOS 10

int main(void)
{
    float notas[NRO_ALUNOS][NRO_NOTAS];
    int lin, col;
    float valor,soma,media;

    for (lin=0;lin<NRO_ALUNOS;lin++)
    {
        for (col=0;col<NRO_NOTAS;col++)
        {
            printf("Nota %d do aluno
                    %d: ",col+1,lin+1);
            scanf("%f",&valor);
            notas[lin][col] = valor;
        }
    }
}
```

```
for (lin=0;lin<NRO_ALUNOS;lin++)
{
    soma = 0;
    for (col=0;col<NRO_NOTAS;col++)
    {
        soma = soma + notas[lin][col];
    }
    media = soma / NRO_NOTAS;
    printf("Aluno %d =
           %.2f\n",lin+1,media);
}
}
```

Linguagem C – Matriz em Subprogramas

- Em linguagem C a passagem de uma matriz para um subprograma pode ser por referência ou parâmetro.
- A matriz dever ser passada informando-se pelo menos o número máximo de colunas.
- Exemplo:

```
#include <stdio.h>

void alt_value(int mat[][10],int linha,int coluna,int valor)
{
    mat[linha][coluna] = valor;
}

int main(void)
{
    int mat[5][10];
    alt_value(mat,1,1,10);
    return 0;
}
```

Linguagem C – Matriz em Subprogramas

- Em linguagem C a passagem de uma matriz para um subprograma pode ser por referência ou parâmetro.
- A matriz deve ser passada informando-se pelo menos o número máximo de colunas.
- Exemplo:

Ou informar também o número máximo de linhas

```
#include <stdio.h>

void alt_value(int mat[5][10], int linha, int coluna, int valor)
{
    mat[linha][coluna] = valor;
}

int main(void)
{
    int mat[5][10];
    alt_value(mat, 1, 1, 10);
    return 0;
}
```

Linguagem C – Matriz em Subprogramas

- Uma outra abordagem é transformar a matriz multidimensional em uma unidimensional (um vetor) e acessar cada posição deslocando-se um número N de colunas por vez.
- A posição no vetor é obtida com a combinação da posição da linha e coluna. Ex: $pos = coluna + linha * MaxColunas$
- Exemplo:

```
#include <stdio.h>
#define LIN 5
#define COL 10

void alt_value(int mat[], int linha, int coluna, int valor)
{
    int pos = coluna + linha * COL;
    mat[pos] = valor;
}

int main(void)
{
    int mat[LIN*COL];
    alt_value(mat, 1, 1, 10);
    return 0;
}
```



MATRIZ - Exercício

- Escrever um algoritmo que leia duas matrizes e calcule a soma delas armazenando em uma terceira matriz. **Apresente a matriz resultante.**



MATRIZ - Exercício

- Escrever um algoritmo para o seguinte problema:

Ler uma matriz A de dimensão $N \times N$ ($N \leq 50$) e verificar se a matriz é simétrica. Escrever a matriz e a mensagem de resposta

Matrize

FAMM

Introdução à Teoria da Computação

Rosário Minghim

Apoio e revisão: Carlos Elias A. Zampieri

Danilo Medeiros Eler

Rogério Eduardo Garcia

Renato Rodrigues