



Universidade Federal Rural de Pernambuco
(UFRPE)
- Unidade Acadêmica de Garanhuns-

Matrizes

Introdução a Linguagem C Parte VI

Prof. Priscilla Kelly Machado Vieira

Apresentação do Capítulo

2

- Introdução
- Vetores Bidimensionais
- Passagem de Matrizes para Funções
- Matriz Representada por um Vetor Simples
- Matriz Representada por um Vetor de Ponteiros
- Exercícios

Introdução

3

- Vetores: Conjunto unidimensional de dados
- Bidimensional, multidimensional
- Matrizes

```
float mat[3][4];
```

	Coluna 0	Coluna 1	Coluna 2	Coluna 3
Linha 0	a[0][0]	a[0][1]	a[0][2]	a[0][3]
Linha 1	a[1][0]	a[1][1]	a[1][2]	a[1][3]
Linha 2	a[2][0]	a[2][1]	a[2][2]	a[2][3]

Matriz Estática

```
float mat[3][4] =  
{ {5,0,8,9},  
  {4,6,8,2},  
  {2,4,7,8} };
```

```
mat[l][c];
```

	Coluna 0	Coluna 1	Coluna 2	Coluna 3
Linha 0	5	0	8	9
Linha 1	4	6	8	2
Linha 2	2	4	7	8

Passagem de Matriz para Funções

5

```
void f (... , int (*mat)[3], ...);  
void f (... , int mat [ ][3], ...);
```

Matrizes Dinâmicas

6

- Sempre que possível usar estática
- C só permite alocar dinamicamente conjuntos unidimensionais
- Criar abstrações conceituais com vetores para representar conjuntos bidimensionais
 - Matriz representada por um vetor simples
 - Matriz representada por um vetor de ponteiros

Matriz Representada por um Vetor Simples

7

- As primeiras posições do vetor para armazenar os elementos da primeira linha, seguidos dos elementos da segunda linha, e assim por diante.
- Estratégia de acesso

a	b	c	d
e	f	g	h
i	j	k	l

a	b	c	d	e	f	g	h	i	j	k	l
---	---	---	---	---	---	---	---	---	---	---	---

`mat[i][j]`
`v[k], k=i*n+j` (n é o
número de colunas)

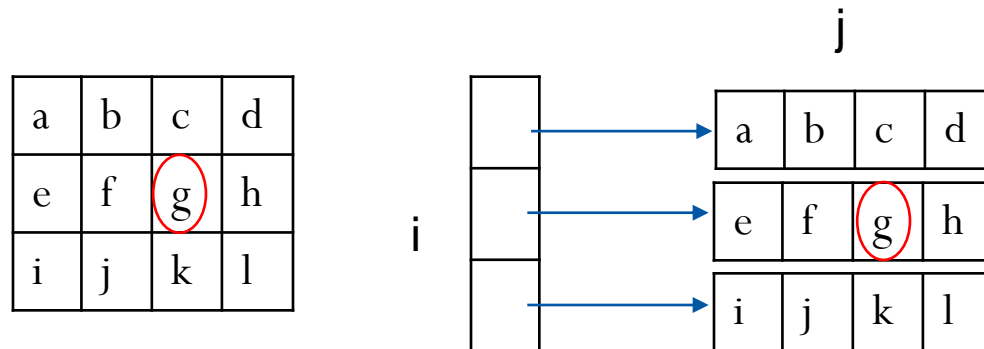
$$k=1*4+2=6$$

```
float *mat  
mat = (float*) malloc  
(m*n*sizeof(float));
```

Matriz Representada por um Vetor de Ponteiros

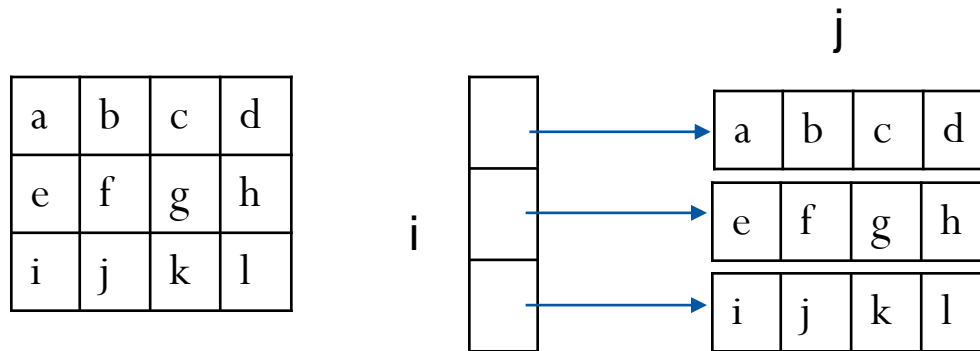
8

- Cada linha da matriz é representada por um vetor independente
- Cada elemento armazena o endereço do primeiro elemento da linhas
 - Representada por um vetor de vetores, ou vetor de ponteiros



Matriz Representada por um Vetor de Ponteiros

9

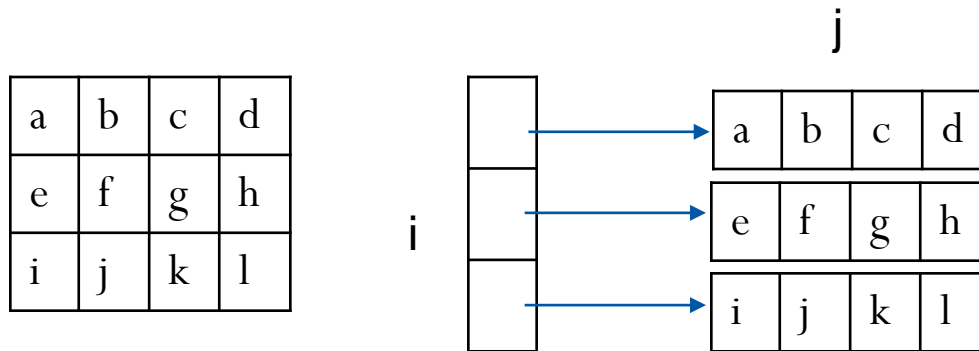


`mat [i][j];`

```
int i;  
float **mat; // matriz representada por um vetor de ponteiros  
mat = (float**) malloc (m*sizeof(float*));  
for (i=0; i<m, i++)  
    mat [i] = (float*) malloc (n*sizeof(float));
```

Matriz Representada por um Vetor de Ponteiros

10



```
int i;  
for (i=0; i<m, i++)  
    free (mat[i]);  
free (mat);
```

Lab 07

11

- Implemente as duas versões vistas para alocação dinâmica de uma matriz e determine a subtração dos elementos de duas matrizes
 - Utilize funções
 - A ordem da matriz deve ser passada pelo usuário, tal como seus elementos