

# Arranjos Bidimensionais

Prof. Fernando Figueira  
(adaptado do material do Prof. Rafael Beserra Gomes)

UFRN

Material compilado em 13 de outubro de 2025.  
Licença desta apresentação:



<http://creativecommons.org/licenses/>

## Arranjos Bidimensionais

# Arranjos Bidimensionais

- **Arranjos (array)**: conjunto de elementos identificáveis por um índice
- Arranjos unidimensionais: **vetores** (aula anterior)
- Arranjos bidimensionais: **matrizes**

# Arranjos Bidimensionais

Representações de matrizes:

■ Matematicamente:

$$M = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{1m} \\ \dots & & & \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix}$$

os elementos são indexados por dois índice ( $M_{ij}$ ) e esses começam do índice 1

# Arranjos Bidimensionais

Representações de matrizes:

- Computacionalmente:

- **Em geral** há um armazenamento contíguo na memória<sup>1</sup>
- Os elementos são indexados por dois índices (geralmente  $m[i][j]$ )
- O usual é primeiro índice para linha e segundo índice para coluna!
- Geralmente começam do índice **0**

---

<sup>1</sup>Se a alocação da matriz for dinâmica, há a possibilidade de alocar linhas diferentes em regiões diferentes da memória.

# Representação de matrizes na memória

$$M = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

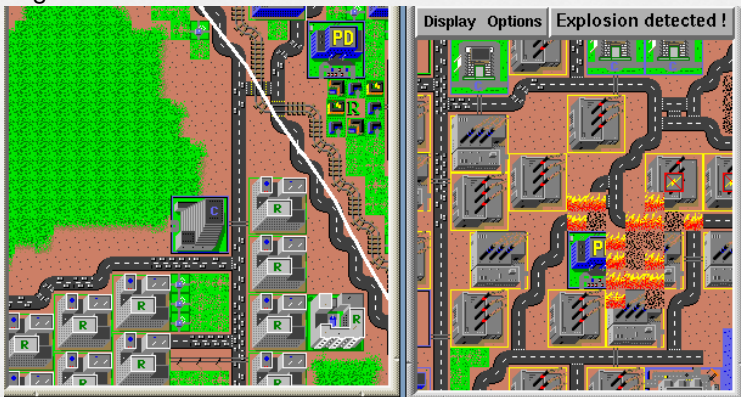
A matriz M pode ser representada da seguinte forma na memória:

Endereço									valor	tipo	identificação
0xbffff22c	0	0	0	0	0	1	0	1	5	inteiro curto	valorIndice
0xbffff22d	0	1	0	0	0	0	1	0	B	caractere	letra1
0xbffff22e	0	1	0	0	0	0	1	1	C	caractere	letra2
0xbffff22f	0	0	0	0	0	0	0	1	1	inteiro curto	M[0][0]
0xbffff230	0	0	0	0	0	0	1	0	2	inteiro curto	M[0][1]
0xbffff231	0	0	0	0	0	0	1	1	3	inteiro curto	M[0][2]
0xbffff232	0	0	0	0	0	1	0	0	4	inteiro curto	M[1][0]
0xbffff233	0	0	0	0	0	1	0	1	5	inteiro curto	M[1][1]
0xbffff234	0	0	0	0	0	1	1	0	6	inteiro curto	M[1][2]
0xbffff235	0	0	1	1	0	0	1	0	2	caractere	letra1

# Aplicações

# Aplicações de matrizes

## ■ Jogos





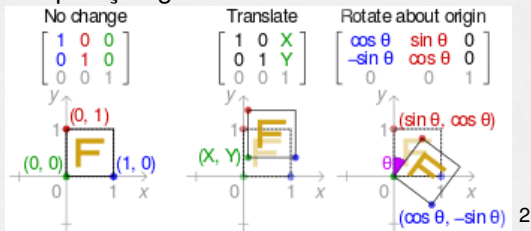
# Aplicações de matrizes

## ■ Jogos



# Aplicações de matrizes

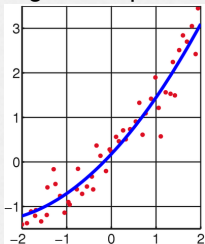
## ■ computação gráfica:



<sup>2</sup>By Cmglee - Own work, CC BY-SA 3.0,  
<https://commons.wikimedia.org/w/index.php?curid=35180401>

# Aplicações de matrizes

- resolução de outros problemas matemáticos, exemplo: regressão por mínimos quadrados

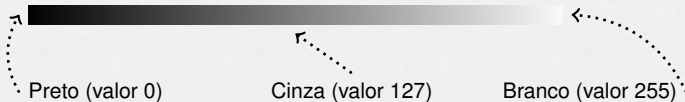
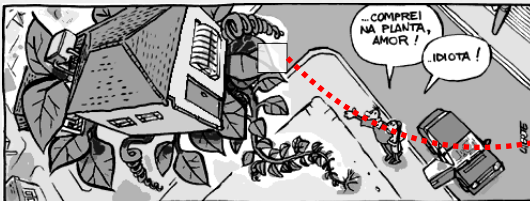


Regressão linear:

$$\begin{bmatrix} n & \sum x_i \\ \sum x_i & \sum x_i^2 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum x_i y_i \end{bmatrix}$$

# Aplicações de matrizes

## ■ imagens digitais



# Aplicações de matrizes

- qualquer conjunto de dados  $n$ -dimensionais
  - Exemplo: um conjunto de  $n$  coordenadas no plano cartesiano pode ser representada em uma matriz  $n \times 2$  ou  $2 \times n$ :

$$\begin{bmatrix} 7 & 2 \\ 5 & 3 \\ 2 & 1 \\ 4 & 1 \end{bmatrix}$$

ou

$$\begin{bmatrix} 7 & 5 & 2 & 4 \\ 2 & 3 & 1 & 1 \end{bmatrix}$$

## Matrizes em C

# Declarando uma matriz em C

Opções:

```
1 #include <stdio.h>
2
3 int main() {
4
5     int matriz1[5][8];
6     int m[][4] = {{1, 2, 3, 4},{6, 7, 8, 9},{11, 12, 13, 14}};
7
8     return 0;
9 }
```

# Acesso ao elemento

Basta identificar o elemento usando seus **índices** entre [] (lembre-se de que começa com 0):

```
1 #include <stdio.h>
2
3 int main() {
4
5     int m[][4] = {{1, 2, 3, 4},{6, 7, 8, 9},{11, 12, 13, 14}};
6
7     printf("%d\n", m[2][3]);
8
9     return 0;
10 }
```



a maioria dos programadores usam linha como primeiro índice e coluna como segundo índice; melhor então se acostumar a pensar dessa forma



## Exemplo 1

- Escrever em uma matriz  $5 \times 5$  os seguintes valores:

0	1	2	3	4
1	2	3	4	5
2	3	4	5	6
3	4	5	6	7
4	5	6	7	8

```
1 #include <stdio.h>
2
3 int main() {
4
5     int m[5][5];
6
7     //algoritmo para preencher a matriz
8
9     return 0;
10 }
```

```
1 for(i = 0; i < 5; i++) {           //for i
2     for(j = 0; j < 5; j++) {       //for j
3         m[i][j] = i+j;
4     }
5     printf("\n");
6 }
```

$i \downarrow \rightarrow j$	0	1	2	3	4
0	0	1	2	3	4
1	?	?	?	?	?
2	?	?	?	?	?
3	?	?	?	?	?
4	?	?	?	?	?

```
1 for(i = 0; i < 5; i++) {           //for i
2     for(j = 0; j < 5; j++) {       //for j
3         m[i][j] = i+j;
4     }
5     printf("\n");
6 }
```

$i \downarrow \rightarrow j$	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	5
2	?	?	?	?	?
3	?	?	?	?	?
4	?	?	?	?	?

```
1 for(i = 0; i < 5; i++) {           //for i
2     for(j = 0; j < 5; j++) {       //for j
3         m[i][j] = i+j;
4     }
5     printf("\n");
6 }
```

i ↓ → j	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	5
2	2	3	4	5	6
3	?	?	?	?	?
4	?	?	?	?	?

```
1 for(i = 0; i < 5; i++) {           //for i
2     for(j = 0; j < 5; j++) {       //for j
3         m[i][j] = i+j;
4     }
5     printf("\n");
6 }
```

i ↓ → j	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	5
2	2	3	4	5	6
3	3	4	5	6	7
4	?	?	?	?	?

```
1 for(i = 0; i < 5; i++) {           //for i
2     for(j = 0; j < 5; j++) {       //for j
3         m[i][j] = i+j;
4     }
5     printf("\n");
6 }
```

i ↓ → j	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	5
2	2	3	4	5	6
3	3	4	5	6	7
4	4	5	6	7	8

## Exemplo 2



## ■ Ler do usuário uma matriz 5 × 5:

```
1 #include <stdio.h>
2
3 int main() {
4
5     int m[5][5];
6     int i, j;
7
8     //leitura
9     for(i = 0; i < 5; i++) {
10         for(j = 0; j < 5; j++) {
11             scanf("%d", &m[i][j]);
12         }
13     }
14     //escrita
15     for(i = 0; i < 5; i++) {
16         for(j = 0; j < 5; j++) {
17             printf("%d ", m[i][j]);
18         }
19         printf("\n");
20     }
21     return 0;
22 }
```



## Exercício em sala

Declare uma matriz  $5 \times 5$  e preencha com uma matriz triangular superior. Depois escreva os valores da matriz na tela.

1	1	1	1	1
0	1	1	1	1
0	0	1	1	1
0	0	0	1	1
0	0	0	0	1



## Exercício em sala

Declare uma matriz  $5 \times 5$  e preencha com a seguinte matriz de tal forma que:

- A primeira linha é 1, 2, 3, 4, 5
- Qualquer outro número da matriz, exceto da última coluna, é igual à soma do número acima com o número à nordeste
- O número da última coluna (exceto o da primeira linha) é igual ao da penúltima coluna

Depois escreva os valores da matriz na tela.

1	2	3	4	5
3	5	7	9	9
8	12	16	18	18
20	30	34	36	36
50	64	70	72	72