

Algoritmos e Programação : Matrizes

Prof. Ovídio Francisco



Plano de aula

Introdução

Dimensões

Matrizes em C

Exercícios

Referências

Exercícios

Introdução

O contexto

É comum que um software precise trabalhar com dados tabulares, ou seja, uma organização onde os dados estão dispostos em linhas e colunas.

Visto essa necessidade, a maioria das linguagens oferecem formas de trabalhar com essas estruturas, as matrizes.

Uma matriz é semelhante a um array, porém com duas dimensões. Podemos então entender uma matriz como uma tabela que possui altura e largura.

Onde é utilizada?

O contexto

O contexto

- As matrizes ou arrays bidimensionais são estruturas de dados tabulares, com linhas e colunas.

	0	1	2	3	4
0					
1					
2					
3					
4					

- Semelhante aos arrays, uma matriz usa índices para acessar um elemento.
- Usa dois índices para acessar um elemento, um para a linha e outro para a coluna.

O contexto

	0	1	2	3	4
0	255	120	80	20	72
1	212	23	32	65	39
2	72	49	46	22	14
3	57	15	69	21	33
4	78	89	32	12	76

- O elemento que está na coluna **3**, linha **2** é **22**.

Onde é utilizada

Matrizes são amplamente utilizadas no desenvolvimento de softwares. Por exemplo:

- Tabelas de uso geral
- Criação de gráficos
- Aplicações matemáticas
- Redes neurais artificiais
- Algoritmos de busca
- Path Finding (Encontrar caminhos)
- Similaridade entre objetos

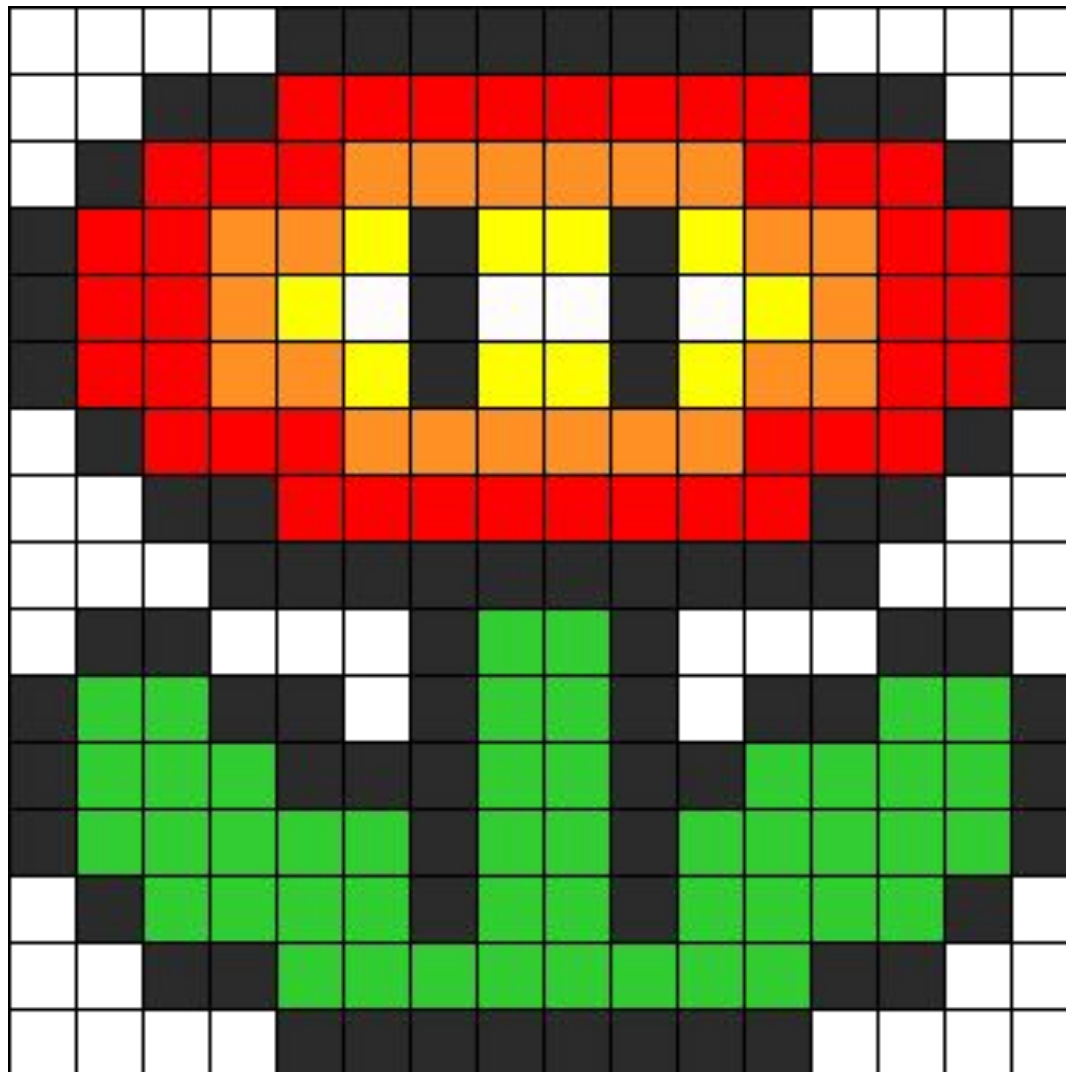
Onde é utilizada

	DS	Química	Mecânica	Nutrição	Farmácia
Tatuí	255	120	80	20	72
Cesário Lange	212	23	32	65	39
Tietê	72	49	46	22	14
Nárnia	57	15	69	21	33
Sorocaba	78	89	32	12	76

Onde é utilizada

255	255	255	255	0	0	0	0	0	0	0	0	255	255	255	255
255	255	0	0	120	120	120	120	120	120	120	120	0	0	255	255
255	0	120	120	120	80	80	80	80	80	80	120	120	120	0	255
0	120	120	80	80	20	0	20	20	0	20	80	80	120	120	255
0	120	120	80	20	20	0	20	20	0	20	20	80	120	120	255
0	120	120	80	80	20	0	20	20	0	20	80	80	120	120	255
255	255	0	0	120	120	120	120	120	120	120	120	0	0	255	255
255	255	255	255	0	0	0	0	0	0	0	0	255	255	255	255
0	120	120	80	20	20	0	20	20	0	20	20	80	120	120	255
0	120	120	80	20	20	0	20	20	0	20	20	80	120	120	255
0	120	120	80	80	20	0	20	20	0	20	80	80	120	120	255
255	255	0	0	120	120	120	120	120	120	120	120	0	0	255	255
255	255	255	255	0	0	0	0	0	0	0	0	255	255	255	255
0	120	120	80	20	20	0	20	20	0	20	20	80	120	120	255
0	120	120	80	80	20	0	20	20	0	20	80	80	120	120	255
255	255	255	255	0	0	0	0	0	0	0	0	255	255	255	255

Onde é utilizada



Onde é utilizada

	Tatuí	Cesário	Tietê	Nárnia	Sorocaba
Tatuí	0	19	35	25	63
Cesário	19	0	44	38	87
Tietê	35	44	0	15	81
Nárnia	25	38	15	0	72
Sorocaba	63	87	81	72	0

Onde é utilizada

	Blade Runner	Jurassic Park	O Senhor dos Anéis	Matrix	Godzilla
Blade Runner	0	19	35	25	63
Jurassic Park	19	0	44	38	87
O Senhor dos Anéis	35	44	0	15	81
Matrix	25	38	15	0	72
Godzilla	63	87	81	72	0

Criando matrizes em JavaScript

Criando matrizes em JavaScript

Uma matriz em JavaScript pode ser entendida como um array de arrays. Por exemplo podemos imaginar um array onde cada elemento também é um array.

Devemos então inicialmente criar um array para em seguida atribuir a cada elemento um novo array.

Veja o exemplo:

```
// Declaração de uma array  
var matrix = [];
```

```
// Atribuição de novo array para cada elemento  
for (var i=0; i<5; i++) {  
    matrix[i] = [];  
}
```

Acessando elementos de uma matriz

Acessando elementos de uma matriz

Um elemento da matriz pode ser acessado por meio de índices:

Exemplo:

```
matrix[2][3] = 8;
```

```
var a = matrix[2][3];
```

Acessando elementos de uma matriz

Para percorrer por todos os elementos de uma matriz, você pode usar *loops* aninhados.

Exemplo:

```
for (var i=0; i<5; i++) {  
    for (var j=0; j<5; j++) {  
        matrix[i][j] = alguma_coisa;  
    }  
}
```

Acessando elementos de uma matriz

Naturalmente, é possível ler os valores de um elemento por meio do comando `scanf`.

Exemplo:

```
matrix[i][j] = prompt("Digite ai: ");
```

```
matrix[3][6] = prompt("Digire ai: ");
```

Para saber mais...

- <https://www.javatpoint.com/c-array>
- https://www.tutorialspoint.com/cprogramming/c_arrays.htm
- <https://www.programiz.com/c-programming/c-arrays>
- <https://www.geeksforgeeks.org/arrays-in-c-cpp/>
- <http://linguagemc.com.br/vetores-ou-arrays-em-linguagem-c/>
- https://www.inf.ufpr.br/cursos/ci067/Docs/NotasAula/notas-18_Arrays.html

Exercícios



Exercícios

1. Crie um programa que cria uma matriz 10x10 que contém todos os números entre 0 e 99.
2. Crie um programa que cria uma matriz 5x5, e preenche todos os elementos com números informados pelo usuário.
3. A partir do exercício 2, faça com que a matriz seja exibida na tela. (Pode usar uma tabela para que os elementos fiquem alinhados.)
4. A partir do exercício 2, calcule a somatória da diagonal principal da matriz.
5. A partir do exercício 2, permita que o usuário informe a linha e coluna de elemento a ser exibido.
6. Crie uma matriz 10x10 onde cada elemento é a soma de seus índices.
7. Crie uma matriz 10x10 onde cada elemento é o dobro do anterior.

Exercícios

1. Crie uma matriz 10x10 com todos os números entre 0 e 99 e em seguida copie todos os elementos para um vetor.
2. Crie um vetor com 25 elementos com os números entre 0 e 24 e em seguida copie todos os elementos para uma matriz quadrada.
3. (Desafio) Crie o jogo caça-palavras. Sugestão: Um jogador informa algumas palavras que serão inseridas em uma matriz de letras. Em seguida outro usuário deve encontrá-las.