

Programação Estruturada

Aula 12 - Arrays: Ordenação, Iteração e Matrizes

Videoaula 03: Matrizes



Videoaula 03: Matrizes

Você lembra o que são matrizes? Elas são elementos matemáticos que oferecem uma solução simples e poderosa à representação dos dados e das relações existentes entre eles. Por exemplo, quando vemos uma tabela, temos, implicitamente, uma matriz de dados. Vamos usar um exemplo que você já deve ter visto em outra disciplina, Matemática Aplicada.

Figura 1 - Tabela de Distâncias

	0	1	2	3	4	5	
CIDADES	Maceió	Manaus	Natal	Palmas	Porto Alegre	Porto Velho	
0	Maceió	0	5.491	572	1.851	3.572	4.505
1	Manaus	2.779	0	5.985	4.141	4.563	901
2	Natal	434	2.765	0	2.345	4.066	4.998
3	Palmas	1.383	1.509	1.527	0	2.747	1.712
4	Porto Alegre	2.775	3.132	3.172	2.222	0	3.662
5	Porto Velho	3.090	761	3.179	1.711	2.706	0

Distância rodoviária Distância aérea

"Essa tabela é um exemplo de matriz composta por dados simples, as distâncias entre pares de capitais, através dos quais fica fácil obter a distância rodoviária e aérea entre duas capitais quaisquer. Dizemos que uma matriz desse tipo possui duas dimensões, da mesma maneira que a tabela correspondente é chamada de tabela de duas entradas, pois relaciona dois grupos de objetos. No caso do nosso exemplo, temos na realidade o mesmo grupo de objetos (as capitais), sendo considerado nas duas dimensões (como cidades de origem e como cidades de destino)." Assim sendo, temos uma matriz quadrada 6 x 6.

Em JavaScript, podemos definir essa matriz através de um array de arrays. Podemos considerar esse tipo de array como um array bidimensional, pois são necessários dois índices para percorrer seu conteúdo. Se ele for tridimensional, você precisará de três índices e assim por diante. Por simplicidade, nesta aula, trabalharemos apenas com matrizes, ou seja, arrays bidimensionais, mas tenha em mente que podemos ter arrays de várias dimensões.

Quando trabalhamos com matrizes, é comum chamarmos os índices de linhas e colunas. No caso, é comum considerarmos que o primeiro índice indexa as linhas e o segundo índice indexa as colunas. Lembre-se que continuamos trabalhando com arrays. Por esse motivo, os índices variam de 0 ao tamanho da dimensão menos um. Veja o exemplo:

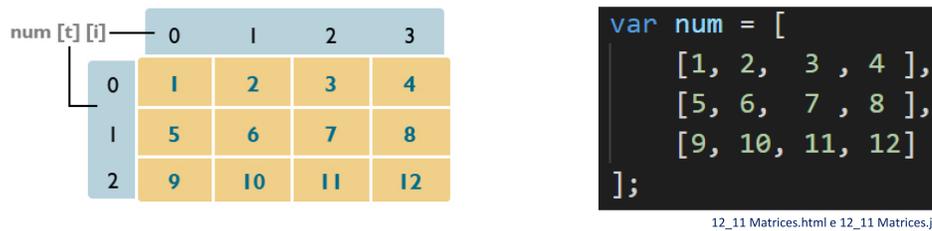
Figura 2 - Matrizes e Índices

num [t] [i]	0	1	2	3
0	1	2	3	4
1	5	6	7	8
2	9	10	11	12

Temos uma matriz com 3 linhas e 4 colunas, ou seja, uma matriz 3x4. Por esse motivo, o índice da primeira dimensão t do array, que representa as linhas, varia entre 0 e 2, e o índice da segunda dimensão i do array, que representa as colunas, varia entre 0 e 3.

Veja agora como fica a declaração dessa matriz em JavaScript.

Figura 3 - Declarando Matrizes em JavaScript

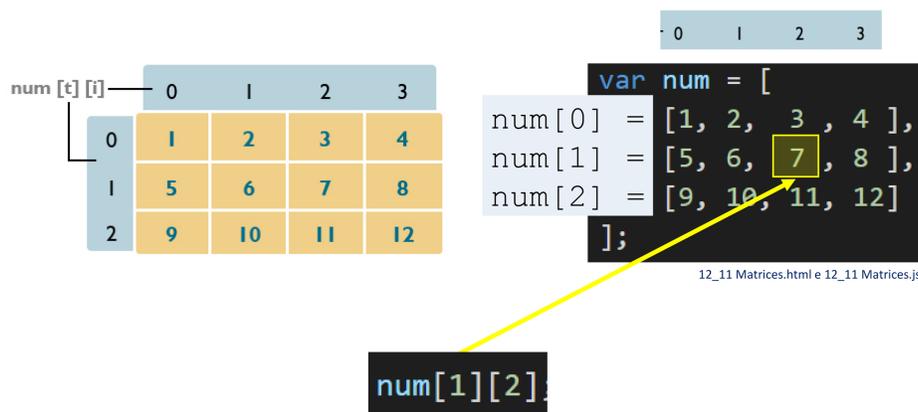


Você consegue fazer um paralelo entre a imagem e o código fonte? Perceba que temos um array chamado `num` cujos elementos são arrays. Para facilitar o entendimento, vamos colocar uma indicação dos elementos do array `num` (Figura 4).

Note que o elemento `num[0]` é o array contendo 1, 2, 3 e 4. De maneira similar, o elemento `num[1]` é o array contendo 5, 6, 7 e 8. Por fim, o elemento `num[2]` é o array contendo 9, 10, 11 e 12.

Agora, você pode acessar cada um dos elementos desses arrays simplesmente usando o índice da coluna. Por exemplo, o elemento 7, marcado em amarelo no slide (Figura 4), é o elemento de índice 2 do array `num[1]`, ou seja, o elemento `num[1][2]`. Simples, não é?

Figura 4 - Acessando Elementos da Matriz



Para darmos um exemplo de uso prático de matrizes em programação, vamos agora voltar à tabela que apresenta as distâncias entre pares de capitais.

Figura 5 - Tabela de Distâncias com Índices

	0	1	2	3	4	5	
CIDADES	Maceió	Manaus	Natal	Palmas	Porto Alegre	Porto Velho	
0	Maceió	0	5.491	572	1.851	3.572	4.505
1	Manaus	2.779	0	5.985	4.141	4.563	901
2	Natal	434	2.765	0	2.345	4.066	4.998
3	Palmas	1.383	1.509	1.527	0	2.747	1.712
4	Porto Alegre	2.775	3.132	3.172	2.222	0	3.662
5	Porto Velho	3.090	761	3.179	1.711	2.706	0

Distância rodoviária

Distância aérea

Note que temos novamente linhas e colunas, representadas pelo primeiro e segundo índices, respectivamente. Na linha 0, temos as distâncias de Maceió, na linha 1, as distâncias de Manaus, indo até a linha 5, onde temos as distâncias de Porto Velho. Em todas as linhas, teremos as distâncias na coluna 0 de Maceió, na coluna 1 de Manaus, que segue até a coluna 5, de Porto Velho. Você notou que sempre que a linha i é igual à coluna j , temos a distância 0? Por exemplo, note que o elemento na linha 2 e na coluna 2, corresponde à distância de Natal para Natal, que, obviamente, é de 0 km.

Existe outra peculiaridade em relação a essa tabela. Você consegue ver alguma relação entre as linhas, as colunas e o tipo de distância na tabela? Pois é, sempre que a linha i for menor que a coluna j , temos a distância rodoviária. Assim, o elemento na linha 0 e na coluna 2 corresponde à distância rodoviária entre Maceió e Natal.

E a outra relação é que sempre que a linha i é maior que a coluna j , temos a distância aérea, como ocorre com o elemento da linha 2 e coluna 0, a qual corresponde à distância aérea entre Natal e Maceió.

Resumindo, dados um identificador de linha i e um identificador de coluna j , temos que:

- Se $i=j$, então a distância é 0;
- Se $i<j$, então a distância é rodoviária;
- E se $i>j$, então a distância é aérea.

Agora, veja como podemos definir no código JavaScript a matriz correspondente à tabela de distâncias.

Figura 6 - Tabela de Distâncias em JavaScript

```
var distancias = [
  [0, 5491, 572, 1851, 3572, 4505],
  [2779, 0, 5985, 4141, 4563, 901],
  [434, 2765, 0, 2345, 4066, 4998],
  [1383, 1509, 1527, 0, 2747, 1712],
  [2775, 3132, 3172, 2222, 0, 3662],
  [3090, 761, 3179, 1711, 2706, 0]
];
```



Iremos desenvolver uma página que permitirá ao usuário escolher dois municípios entre os listados na tabela, o tipo de distância (rodoviária ou aérea), e a página retornará o valor da distância. Vamos lá?

Código 1 - 12_12 Distancias.html

```
1 <html >
2   <head>
3     <meta charset="UTF-8" />
4     <title>Programação Estruturada - Aula 12</title>
5   </head>
6   <body>
7     <noscript>Seu navegador não suporta JavaScript ou ele está desabilitado.</noscript>
8
```

```

9     <h1>Distâncias</h1>
10
11    Cidade 1:
12    <select id="cidade1">
13        <option value=0>Maceió</option>
14        <option value=1>Manaus</option>
15        <option value=2>Natal</option>
16        <option value=3>Palmas</option>
17        <option value=4>Porto Alegre</option>
18        <option value=5>Porto Velho</option>
19    </select>
20
21    Cidade 2:
22    <select id="cidade2">
23        <option value=0>Maceió</option>
24        <option value=1>Manaus</option>
25        <option value=2>Natal</option>
26        <option value=3>Palmas</option>
27        <option value=4>Porto Alegre</option>
28        <option value=5>Porto Velho</option>
29    </select>
30
31    Tipo de Distância:
32    <select id="distancia">
33        <option value=0>Rodoviária</option>
34        <option value=1>Aérea</option>
35    </select>
36
37    <button onclick="calcular()">OK</button><br>
38
39    <p id="resultado"></p>
40
41    <script src="12_12 Distancias.js"></script>
42 </body>
43 </html>
44

```

```

1 var distancias = [
2   [0 , 2779, 434 , 1383, 2775, 3090],
3   [5491, 0 , 2765, 1509, 3132, 761 ],
4   [572 , 5985, 0 , 1527, 3172, 3179],
5   [1851, 4141, 2345, 0 , 2222, 1711],
6   [3572, 4563, 4066, 2747, 0 , 2706],
7   [4505, 901 , 4998, 1712, 3662, 0 ]
8 ];
9
10 function calcular() {
11     var cidade1 = document.getElementById("cidade1");

```

```

12 var indice1 = cidade1.options[cidade1.selectedIndex].value;
13 var nome1 = cidade1.options[cidade1.selectedIndex].text;
14
15 var cidade2 = document.getElementById("cidade2");
16 var indice2 = cidade2.options[cidade2.selectedIndex].value;
17 var nome2 = cidade2.options[cidade2.selectedIndex].text;
18
19 var distancia = document.getElementById("distancia");
20 var valorD = distancia.options[distancia.selectedIndex].value;
21 var textoD = distancia.options[distancia.selectedIndex].text;
22
23 var maior_indice = Math.max(indice1, indice2);
24 var menor_indice = Math.min(indice1, indice2);
25
26 var distancia = 0;
27 if (valorD == 0) {
28     // Distância Rodoviária (i < j)
29     distancia = distancias[menor_indice][maior_indice];
30 } else {
31     // Distância Aérea (i > j)
32     distancia = distancias[maior_indice][menor_indice];
33 }
34
35 var resposta = "A distância " + textoD;
36 resposta = resposta + " entre " + nome1;
37 resposta = resposta + " é " + nome2;
38 resposta = resposta + " é de " + distancia + "Km";
39
40 document.getElementById("resultado").innerHTML = resposta;
41 }
42
43 //var i = Number(document.getElementById("cidade1").value);
44

```

Primeiro, vamos conhecer o HTML, nele tem coisas diferentes do que a gente vem usando ao longo da disciplina. Veja que a gente usa duas caixas de seleção, o primeiro select, que tem o id "cidade1" e o outro que tem id "cidade2". E as opções de ambos são iguais, seis opções com valores variando de 0 até 5 com os nomes das cidades. Perceba que a opção 1 da "cidade1" e a opção da "cidade2" são iguais.

Além disso, precisamos de uma terceira caixa de seleção, estão nas linhas 31 a 35, que é a distância que possui valores 0 e 1, **0 para rodoviária** e **1 para aérea**. E quando eu clicar no botão OK, a gente vai chamar a função `calcular`, tá bom?

Então, vamos ver agora como funciona a função `calcular` no JavaScript que estamos utilizando. A primeira coisa que a gente precisa fazer é pegar quais foram as cidades escolhidas. Então, vou pegar o elemento do HTML, "cidade1" que é exatamente aquele elemento que tem o identificador "cidade1" na linha 11 e aí, nas linhas 12 e 13 eu pego, respectivamente, o valor da opção e o texto da opção, como podemos fazer isso?

Figura 7 - Acessando o Valor e o Texto da Opção Selecionada

```
11  var cidade1 = document.getElementById("cidade1");
12  var indice1 = cidade1.options[cidade1.selectedIndex].value;
13  var nome1 = cidade1.options[cidade1.selectedIndex].text;
```

Basta irmos no array de opções que ele tem e pegarmos o `cidade1.selectedIndex`, que é o índice selecionado daquele elemento, então ele vai retornar o índice que foi selecionado e a gente pega o campo valor, que é aquele que a gente colocou nas opções.

Figura 8 - Relembrando a Definição das Opções

```
21  Cidade 2:
22  <select id="cidade2">
23    <option value=0>Maceió</option>
24    <option value=1>Manaus</option>
25    <option value=2>Natal</option>
26    <option value=3>Palmas</option>
27    <option value=4>Porto Alegre</option>
28    <option value=5>Porto Velho</option>
```

São esses valores que aparecem nas linhas 23 a 28 do HTML, então se você selecionar "Maceió", de acordo com o que a gente fez lá na página, você vai retornar o valor 0 e o texto vai retornar exatamente "Maceió". Então a gente vai fazer exatamente a mesma coisa para pegar o índice da "cidade2" e o nome da "cidade2". E também para pegar o valor e o texto da distância, então o `valorD` vai ser 0 ou 1 e o `textoD` vai ser rodoviária ou aérea.

Agora, a gente vai usar esses valores para identificar a distância e construir aquele texto. Como iremos fazer isso? Na linha 26, eu começo dizendo que a distância é 0, então basicamente o que vou fazer primeiro é saber se eu pego a distância rodoviária

ou aérea. A rodoviária é se o `valorD` é 0, e, caso contrário, é aérea.

Então basta trabalharmos com os índices, lembre-se: se eu colocar o menor índice como primeiro e o maior índice como segundo, vou pegar a distância rodoviária. E se eu fizer o contrário, pego a distância aérea. Olha o que eu fiz nas linhas 23 e 24.

Figura 9 - Identificando o Menor e o Maior Índice

```
23   var maior_indice = Math.max(indice1, indice2);
24   var menor_indice = Math.min(indice1, indice2);
```

Aqui, eu peguei o maior índice dos dois índices, `indice1` e `indice2`, quem é o maior e o menor índice. Como fiz isso? Usando a função que conhecemos em aulas anteriores, `max` e `min`.

A função `Math.max` vai retornar o maior valor entre os dois passados e a função `Math.min` vai retornar o menor valor dos dois passados, assim fico sabendo quem é o maior índice e o menor índice. Então agora sei que é para colocar o menor índice como primeiro e o maior índice como segundo, ou vice-versa, dependendo se é a distância rodoviária ou aérea.

Assim, se a distância for rodoviária, sabemos que o `i` é menor que o `j`, então coloco na linha 29 o `menor_indice` como primeiro e o `maior_indice` como segundo. Caso contrário, inverte e sei que o `i` é maior que `j`, então coloco o `maior_indice` primeiro, o `i`, e o `menor_indice` como segundo, o `j`.

Figura 10 - Retornando a Distância Rodoviária ou a Distância Aérea

```
26   var distancia = 0;
27   if (valorD == 0) {
28       // Distância Rodoviária (i < j)
29       distancia = distancias[menor_indice][maior_indice];
30   } else {
31       // Distância Aérea (i > j)
32       distancia = distancias[maior_indice][menor_indice];
33   }
```

Isso faz com que peguemos exatamente a distância rodoviária ou aérea entre as duas cidades. Agora, é só uma questão de construir o texto, logo eu coloco: "A distância" e o `textoD`, ou seja, o texto que está na opção "rodoviária" ou "aérea", entre a primeira cidade e a segunda cidade, que é o `nome1` e o `nome2`, "é de" e coloco a distância que a gente calculou entre as linhas 27 e 33.

Figura 11 - Construindo o Texto a Ser Exibido

```
35 var resposta = "A distância " + textoD;
36 resposta = resposta + " entre " + nome1;
37 resposta = resposta + " e " + nome2;
38 resposta = resposta + " é de " + distancia + "Km";
```

E o resultado disso, como podemos ver na página HTML ao pegarmos outra distância, por exemplo, Porto Alegre para Natal, a distância aérea daqui para lá, e a gente constrói o texto (Figura 12). Ok?

Figura 12 - Saída na Página HTML

Distâncias

Cidade 1: Cidade 2: Tipo de Distância:

A distância Aérea entre Porto Alegre é Natal é de 3172Km

Massa, né? Pois é, com arrays e matrizes podemos escrever programas muito úteis. Não deixe de praticar bastante a utilização dessas estruturas. Para começar, que tal fazer os exercícios propostos a seguir? **Lembre-se:** pratique bastante, começando pelos exercícios desta aula, para se acostumar com essa notação.

Até a próxima aula. Tchau, Tchau!!!