



# Calculadora de Matrizes

UC Estruturas Matemáticas - 2024/1



# Integrantes

Antonio Bruno Figueiredo De Oliveira | RA: 822165499

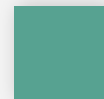
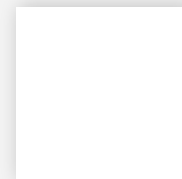
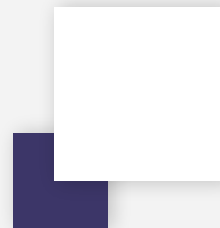
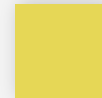
Caio Vieira De Camargo | RA: 8222241862

Juliana Magiero Benedetti | RA: 822229587

Natália Brediks Miltus Marques | RA: 822226393

Pedro Leonardo Rodrigues Ferreira | RA: 822167374

Ruan Vinicius Luz Dos Santos | RA: 8222244748

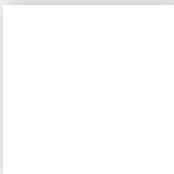


# Matrizes na Computação

As **matrizes** desempenham um papel *crucial na computação*, sendo fundamentais em diversas áreas e aplicações. Elas são essenciais no processamento de imagens e gráficos, incluindo a renderização de objetos 3D e representações geográficas.







Além disso, as matrizes são amplamente utilizadas em álgebra linear, na representação de grafos e em análise numérica, permitindo a resolução eficiente de problemas complexos e o desenvolvimento de algoritmos avançados.

# O projeto: Calculadora de Matrizes



Com base nisso, desenvolvemos uma **calculadora de matrizes** baseada na **web** para realizar operações matemáticas de forma eficiente e precisa.

Esta aplicação permite ao usuário realizar adição, subtração, multiplicação, cálculo de determinantes e inversa de matrizes, destacando as estruturas matemáticas e proporcionando uma **interface intuitiva e funcional** para facilitar o trabalho com matrizes.



# Interface: Menu

## CÁLCULO DE MATRIZES

### Selecione a Operação

☒ Adição   ☐ Subtração   ☐ Multiplicação   ☐ Determinante   ☐ Inversa

### Quantidades desejadas

Quantidade de linhas e colunas:

X

Próximo

# Interface: Inserir elementos

## CÁLCULO DE MATRIZES

Selecione a Operação

☒ Adição   ☐ Subtração   ☐ Multiplicação   ☐ Determinante   ☐ Inversa

Quantidades desejadas

Quantidade de linhas e colunas:

X

### Matrizes

[


]

+

[


]

Menu   Calcular

# Interface: Resultado

## CÁLCULO DE MATRIZES

Resultado: 

4	4
4	4

Menu

# Interface: Avisos

The screenshot shows a web browser window with the address bar displaying `peter94157.github.io/Calculadora_Matriz/Index.html#resultado`. The page title is "CÁLCULO DE MATRIZES". A modal alert box is open, displaying the text "peter94157.github.io diz" and "Preencha os campos em branco" (Fill in the blank fields), with an "OK" button. The main form contains two sections: "Selecione a Operação" (Select the Operation) with radio buttons for "Adição" (selected), "Subtração", "Multiplicação", "Determinante", and "Inversa"; and "Quantidades desejadas" (Desired quantities) with the label "Quantidade de linhas e colunas:" (Number of rows and columns:) and input fields containing "1" and "X" followed by an empty field. A "Próximo" (Next) button is at the bottom.

**CÁLCULO DE MATRIZES**

**Selecione a Operação**

☒ Adição ☐ Subtração ☐ Multiplicação ☐ Determinante ☐ Inversa

**Quantidades desejadas**

Quantidade de linhas e colunas:

1 X

**Próximo**



# Interface: Avisos

The screenshot shows a web browser window with the URL `https://peter94157.github.io/Calculadora_Matrizes/Index.html#resultado`. The page title is "CÁLCULO DE MATRIZES". A modal dialog box is displayed in the center, titled "peter94157.github.io diz", with the message "A matriz deve ser quadrada para calcular o determinante" and an "OK" button. Below the dialog, the interface has a section "Selecione a Operação" with five radio buttons: "Adição", "Subtração", "Multiplicação", "Determinante" (which is selected), and "Inversa". Below this is a section "Quantidades desejadas" with the text "Quantidade de linhas e colunas:" followed by input fields containing "3" and "5" separated by an "X". At the bottom of the form is a "Próximo" button.

← ↻ 🔒 `https://peter94157.github.io/Calculadora_Matrizes/Index.html#resultado` 🔍 ⓘ ☆ ⚙️ ⋮

🔍 Google 🔍 Copilot com GPT-4 📺 YouTube 🌐 Página Inicial | Micr... peter94157.github.io diz Recently viewed - F... 🌐 Home - Canva 📷 Instagram >

**peter94157.github.io diz**  
A matriz deve ser quadrada para calcular o determinante  
OK

## CÁLCULO DE MATRIZES

Selecione a Operação

☐ Adição ☐ Subtração ☐ Multiplicação ☒ Determinante ☐ Inversa

Quantidades desejadas

Quantidade de linhas e colunas:

3 X 5

Próximo

# Metodologia



## Linguagens

Utilização de HTML, CSS e JavaScript.



## Estrutura de Dados

Arrays bidimensionais para representar matrizes.



## Flexibilidade de Entrada

Flexibilidade na especificação das dimensões das matrizes pelo usuário.



## Implementação das Operações

Operações implementadas com estruturas de repetição.

# Operações Realizadas

**01**

## **Adição/Subtração**

Elemento por elemento

**02**

## **Multiplicação**

Combinação linear das linhas e colunas

**03**

## **Determinante**

Utiliza eliminação gaussiana

**04**

## **Inversa**

Método da matriz adjunta

# 01

## Adição/Subtração

Na adição e subtração de matrizes, as operações são realizadas **elemento por elemento**. Isso significa que cada elemento das duas matrizes é somado ou subtraído com seu correspondente, e o **resultado é armazenado em uma nova matriz**.

Esse processo é feito iterando sobre cada elemento das matrizes envolvidas.

```
152 function realizarOperacao(matrizA, matrizB,
153 operacao) {
154     const resultado = [];
155     for (let i = 0; i < matrizA.length; i++) {
156         const row = [];
157         for (let j = 0; j < matrizA[0].length; j++)
158         ) {
159             if (operacao === 'soma') {
160                 row.push(matrizA[i][j] + matrizB[i
161                 ][j]);
162             } else if (operacao === 'subtracao') {
163                 row.push(matrizA[i][j] - matrizB[i
164                 ][j]);
165             } else if (operacao ===
166             'multiplicacao') {
167                 let sum = 0;
168                 for (let k = 0; k < matrizA[0].
169                 length; k++) {
170                     sum += matrizA[i][k] * matrizB
171                     [k][j];
172                 }
173                 row.push(sum);
174             }
175         }
176         resultado.push(row);
177     }
178     return resultado;
179 }
180
181 function exibirResultado(resultado) {
182
183     const divResultado = document.getElementById(
184     'resultado');
```

## 02

# Multiplicação

Na multiplicação de matrizes, cada elemento da matriz resultante é obtido **multiplicando e somando os produtos** dos elementos das linhas da primeira matriz pelas colunas da segunda matriz.

Esse processo é realizado utilizando **loops** para percorrer os elementos das matrizes e realizar os cálculos necessários.

```
152 function realizarOperacao(matrizA, matrizB,
153 operacao) {
154     const resultado = [];
155     for (let i = 0; i < matrizA.length; i++) {
156         const row = [];
157         for (let j = 0; j < matrizA[0].length; j++)
158         ) {
159             if (operacao === 'soma') {
160                 row.push(matrizA[i][j] + matrizB[i
161                 ][j]);
162             } else if (operacao === 'subtracao') {
163                 row.push(matrizA[i][j] - matrizB[i
164                 ][j]);
165             } else if (operacao ===
166             'multiplicacao') {
167                 let sum = 0;
168                 for (let k = 0; k < matrizA[0].
169                 length; k++) {
170                     sum += matrizA[i][k] * matrizB
171                     [k][j];
172                 }
173                 row.push(sum);
174             }
175         }
176         resultado.push(row);
177     }
178     return resultado;
179 }
180
181 function exibirResultado(resultado) {
182
183     const divResultado = document.getElementById(
184     'resultado');
```

### 03

## Determinante

O determinante de uma matriz pode ser calculado de várias maneiras, dependendo do tamanho da matriz. A aplicação utiliza o **método da eliminação de Gauss**, onde a função "calcularDeterminante" converte a matriz em uma **matriz triangular superior** e calcula o determinante como o **produto** dos elementos da **diagonal principal**.

Se um pivô for zero, realiza-se a troca de linhas.

$A_{11}$	$A_{12}$	$A_{13}$	$A_{14}$
0	$A_{22}$	$A_{23}$	$A_{24}$
0	0	$A_{33}$	$A_{34}$
0	0	0	$A_{44}$

# O código

## Cálculo da matriz determinante

```
← main.js
271
272 function calcularDeterminante(matriz) {
273   // Função para calcular o determinante de uma
    matriz
274   const n = matriz.length;
275   if (n !== matriz[0].length) {
276     throw new Error('A matriz deve ser
    quadrada para calcular o determinante')
277   };
278
279   let det = 1;
280   let copy = matriz.map(row => row.slice());
281
```

```
281
282   for (let i = 0; i < n; i++) {
283     let factor = copy[i][i];
284     if (factor === 0) {
285       for (let j = i + 1; j < n; j++) {
286         if (copy[j][i] !== 0) {
287           [copy[i], copy[j]] = [copy[j],
    copy[i]];
288           factor = copy[i][i];
289           det *= -1;
290           break;
291         }
292       }
293       if (factor === 0) {
294         return 0;
295       }
296     }
297
298     det *= factor;
299     for (let j = i; j < n; j++) {
300       copy[i][j] /= factor;
301     }
302
303     for (let j = i + 1; j < n; j++) {
304       factor = copy[j][i];
305       for (let k = i; k < n; k++) {
306         copy[j][k] -= factor * copy[i][k];
307       }
308     }
309   }
310
311   return det;
312
313 }
```

## 04 Inversa

A função "calcularInversa" utiliza o método de **eliminação gaussiana** para encontrar a **inversa** de uma matriz. Esse método transforma a matriz original em uma **matriz identidade** e realiza as mesmas operações na matriz identidade correspondente para obter a **inversa**.

$$\left( \begin{array}{ccc|ccc} a_{11} & a_{12} & a_{13} & 1 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & 0 & 1 & 0 \\ a_{31} & a_{32} & a_{33} & 0 & 0 & 1 \end{array} \right)$$



```

224 function calcularInversa(matriz) {
225
226     // Função para calcular a inversa de uma matriz
227     const n = matriz.length;
228     let identity = [];
229     for (let i = 0; i < n; i++) {
230         identity[i] = [];
231         for (let j = 0; j < n; j++) {
232             identity[i][j] = (i === j) ? 1 : 0;
233         }
234
235         for (let i = 0; i < n; i++) {
236             let factor = matriz[i][i];
237             if (factor === 0) {
238                 for (let j = i + 1; j < n; j++) {
239                     if (matriz[j][i] !== 0) {
240                         [matriz[i], matriz[j]] = [
241                             matriz[j], matriz[i]];
242                         [identity[i], identity[j]] = [
243                             identity[j], identity[i]];
244                         factor = matriz[i][i];
245                         break;
246                     }
247                 }
248                 if (factor === 0) {
249                     alert('Matriz não tem inversa');
250                     throw new Error(
251                         'Matriz não tem inversa');
252                     return
253                 }
254                 for (let j = 0; j < n; j++) {
255                     matriz[i][j] /= factor;
256                     identity[i][j] /= factor;

```

```

253         for (let j = 0; j < n; j++) {
254             matriz[i][j] /= factor;
255             identity[i][j] /= factor;
256         }
257
258         for (let j = 0; j < n; j++) {
259             if (j !== i) {
260                 factor = matriz[j][i];
261                 for (let k = 0; k < n; k++) {
262                     matriz[j][k] -= factor *
263                         matriz[i][k];
264                     identity[j][k] -= factor *
265                         identity[i][k];
266                 }
267             }
268
269             return identity;
270 }

```

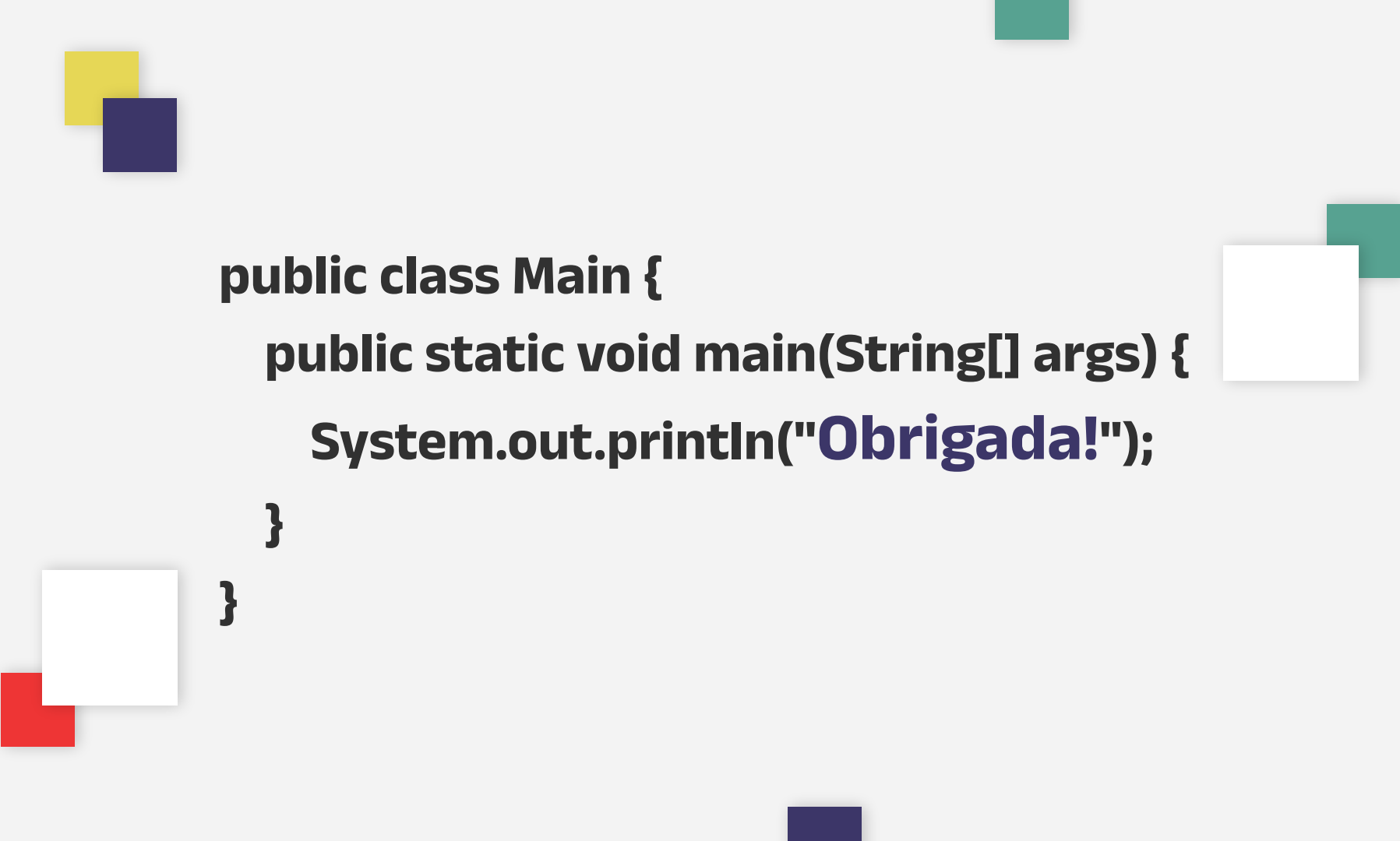
# O código

Cálculo da *matriz inversa*

# Vídeo do funcionamento da Calculadora



[Link](#)



```
public class Main {  
    public static void main(String[] args) {  
        System.out.println("Obrigada!");  
    }  
}
```

```
105     var resultado = [];  
106     switch (operacao) {  
107         case 'soma':  
108             resultado = realizarOperacao(matrizA,  
matrizB, operacao)  
109             break;  
110         case 'multiplicacao':  
111             resultado = realizarOperacao(matrizA,  
matrizB, operacao)  
112             break;  
113         case 'subtracao':  
114             resultado = realizarOperacao(matrizA,  
matrizB, operacao)  
115             break;  
116         case 'determinante':  
117             resultado = calcularDeterminante(  
matrizA);  
118             break;  
119         case 'inversa':  
120             resultado = calcularInversa(matrizA)  
121             break;  
122         default:  
123             alert('operacao inválida');  
124             return;  
125     }  
126  
127     return exibirResultado(resultado);  
128  
129  
130 }
```

# O código

## Identificando o cálculo selecionado e calculando

```

3
4 function criarField() {
5     const calculoSelecioneado = document.
      querySelector('input[name="operacao"]:checked').
      value
6     const linhas = parseInt(document.
      getElementById('qtdLin').value);
7     const colunas = parseInt(document.
      getElementById('qtdCol').value);
8     if (isNaN(linhas) || isNaN(colunas)) {
9         return alert(
      "Preencha os campos em branco")
10    }
11    document.getElementById("Matrizes").innerHTML
      = `<fieldset>
12      <legend>Matrizes</legend>
13
14      <p style="font-size: 80px; width: 20px;">[</p>
15      <div id="matrizA"></div>
16      <p style="font-size: 80px; width: 20px;">]</p>
17      <h1 id="operador"></h1>
18      <p style="font-size: 80px; width: 20px;">[</p>
19      <div id="matrizB"></div>
20      <p style="font-size: 80px; width: 20px;">]</p>
21      </fieldset>`
22
23
24    if (calculoSelecioneado == "determinante" ||
      calculoSelecioneado == "inversa") {
25        excluirSecoes(
      '#Matrizes > fieldset > p:nth-child(6)')
26        excluirSecoes(
      '#Matrizes > fieldset > p:nth-child(7)')
27        excluirSecoes('matrizB')
28    }
29
30 }

```

# 0 código

## Criação do campo para inserir as matrizes

```

36
37 function criarMatriz(id) {
38
39     const linhas = parseInt(document.
40     getElementById('qtdLin').value);
41     const colunas = parseInt(document.
42     getElementById('qtdCol').value);
43     const operacao = document.querySelector(
44     'input[name="operacao"]:checked').value;
45     const divMatriz = document.getElementById(id);
46
47     if (isNaN(linhas) || isNaN(colunas)) {
48         return
49     }
50     if (divMatriz == '') {
51         divMatriz.innerHTML = '';
52     }
53
54     for (let i = 0; i < linhas; i++) {
55         for (let j = 0; j < colunas; j++) {
56             const input = document.createElement(
57             'input');
58             input.type = 'number';
59             input.setAttribute('class',
60             'campoForms');
61             console.log(input)
62             divMatriz.appendChild(input);
63         }
64         divMatriz.appendChild(document.
65         createElement('br'));
66     }

```

```

62     const sinal = document.getElementById(
63     "operador")
64     switch (operacao) {
65         case 'soma':
66             sinal.innerHTML = '+';
67             break;
68         case 'subtracao':
69             sinal.innerHTML = '-';
70             break;
71         case 'multiplicacao':
72             sinal.innerHTML = 'x';
73             break;
74         case 'inversa':
75             sinal.innerHTML = ' '
76             break;
77         case 'determinante':
78             sinal.innerHTML = ' '
79             break;
80         default:
81             alert('erro no simbolo da operação')
82             return;
83     }
84
85     document.getElementById("button").innerHTML
86     = `<button onclick="calcularMatrizes(${operacao})">Calcular</button>`;
87     document.getElementById("next").innerHTML = ""
88     document.getElementById("next").innerHTML =
89     '<button onclick="reload()">Menu</button>'
90     document.querySelector(
91     '#matrizA > input:nth-child(1)').focus()
92

```

# O código

## Criando as matrizes

