

INTRODUÇÃO À CIÊNCIA DA COMPUTAÇÃO I
SCC0221

MULTIPLICAÇÃO DE MATRIZES

1 Introdução

Na matemática, o processo de multiplicar matrizes é muito importante nas áreas de Geometria Analítica e Álgebra Linear, uma vez que possui uma representação geométrica muito sólida quando pensamos em vetores. Apesar de parecer abstrato, o processo de multiplicar duas matrizes é mecânico: facilmente, é possível construir um algoritmo que nos exiba a solução da multiplicação de duas matrizes.

O processo de multiplicação de duas matrizes

A multiplicação de uma matriz A , de n linhas e p colunas, com uma matriz B , de p linhas e q colunas, resultará em uma matriz de n linhas e q colunas. Isso nos mostra uma importante propriedade dessa operação com matrizes: **Na multiplicação entre duas matrizes A e B , o número de colunas de A deve ser igual ao número de linhas de B . Caso contrário, não é possível realizar a multiplicação.**

O processo de cálculo numérico dos elementos da matriz $C = A \times B$ é, de certa forma, simples: **O valor numérico do componente c_{ij} da matriz C é dado pelo produto da i -ésima linha de A com a j -ésima linha de B .**

Por exemplo: Considere duas matrizes $M_{3 \times 3}$ e $N_{3 \times 3}$ dadas a seguir e acompanhe o processo de multiplicação destas.

$$M \times N = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 0 & 2 \\ 1 & 3 & 2 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 2 \\ 1 & 3 & 2 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 \cdot 0 + 2 \cdot 1 + 3 \cdot 0 & 1 \cdot 1 + 2 \cdot 3 + 3 \cdot 0 & 1 \cdot 2 + 2 \cdot 2 + 3 \cdot 1 \\ 2 \cdot 0 + 0 \cdot 1 + 2 \cdot 0 & 2 \cdot 1 + 0 \cdot 3 + 2 \cdot 0 & 2 \cdot 2 + 0 \cdot 2 + 2 \cdot 1 \\ 1 \cdot 0 + 3 \cdot 1 + 2 \cdot 0 & 1 \cdot 1 + 3 \cdot 3 + 2 \cdot 0 & 1 \cdot 2 + 3 \cdot 2 + 2 \cdot 1 \end{bmatrix}.$$

Motivado pelos seus estudos sobre Geometria Analítica, você se encarrega de construir um algoritmo que multiplique duas matrizes. Na página a seguir, é possível encontrar uma representação visual mais didática do passo a passo de como funciona a multiplicação de matrizes, de maneira a esclarecer possíveis dúvidas com relação ao seu processo.

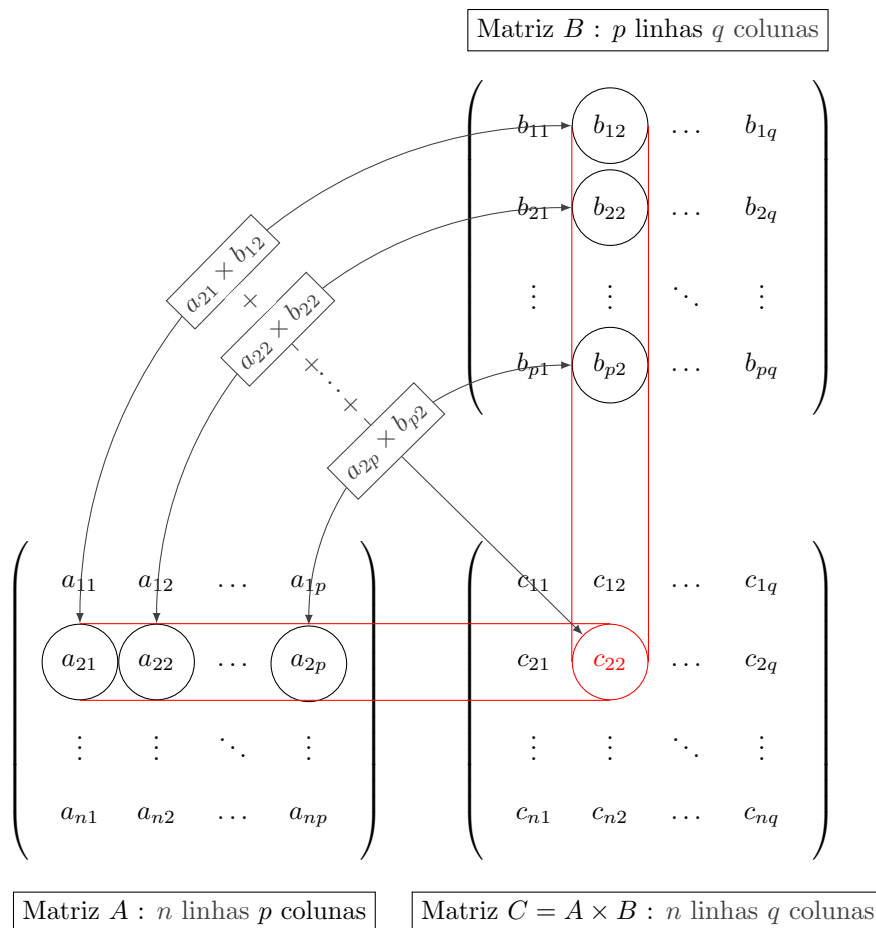


Figura 1: Representação visual da multiplicação de matrizes.

Sobre a geração dos elementos das matrizes

Para esse trabalho, cada elemento das matrizes será gerado através de um gerador de números inteiros pseudoaleatórios, nativamente implementado da biblioteca `stdlib.h`.

A função `rand()` nos permite gerar um inteiro pseudoaleatório baseado em uma semente, ou *seed*, que cria um viés na geração desses números, definida pela função `srand(S)`, onde S é a semente. A semente, então, deve ser definida **antes** de qualquer ocorrência da função `rand()` no código.

- Nessa implementação, utilize a expressão `'rand()%50 - 25'` para gerar um inteiro pseudoaleatório entre -25 e 24.
- Alimente as matrizes na ordem de entrada, ou seja, gere primeiro os números da matriz A e depois os da matriz B . Inverter essa ordem ocasionará em matrizes diferentes (a multiplicação de matrizes não é comutativa) e em resultados finais incorretos.

Exemplo de utilização

```
1  #include <stdlib.h>
2
3  ...
4
5  int semente;
6  scanf("%d", &semente);
7
8  // Definicao da semente de aleatoriedade
9  srand(semente);
10
11 ...
12
13 // Gera um numero aleatorio e armazena na posicao (i,j) da matriz.
14 matrix[i][j] = rand()%50 - 25;
15
```

2 Entrada

Serão dados, de entrada, as dimensões L_A e C_A de uma matriz A (linhas e colunas, respectivamente), separados por um espaço. A seguir será fornecido, em uma nova linha, as dimensões L_B e C_B de uma matriz B , também separados por um espaço. Todos estes são números inteiros. Por fim será dado, em outra linha, um valor inteiro S que representará a *seed* da geração de elementos.

- Os valores L_A , L_B , C_A e C_B devem ser positivos.
- O valor de S deve ser inteiro.

3 Saída

A saída deve conter a impressão da matriz resultante $C = A \times B$, com cada termo separado por espaços. As linhas da matriz também devem ser preservadas na impressão. Confira a seção de **Exemplos de entrada e saída** (4) para melhor noção da formatação esperada.

- Caso não seja possível efetuar a multiplicação (por qualquer que seja o motivo), o programa não deve prosseguir com a operação e nem ler a *seed* (isto é, ele deverá parar a execução antes de lê-la), exibindo na tela a mensagem `Valores invalidos para a multiplicacao\n`. A função `exit(0)` pode te ajudar nessa tarefa.

4 Exemplos de entrada e saída

- Exemplo 1

Entrada

```
1 3 2
2 2 4
3 12
```

Saída

```
1 Linha 0: -93 96 -132 176
2 Linha 1: 25 64 724 -192
3 Linha 2: -68 124 316 42
```

- Exemplo 2

Entrada

```
1 4 3
2 3 6
3 -14
```

Saída

```
1 Linha 0: -349 473 -98 520 -741 -456
2 Linha 1: 399 525 -210 840 -777 -168
3 Linha 2: 859 -75 -86 168 195 444
4 Linha 3: 451 -374 65 -448 759 573
```

- Exemplo 3

Entrada

```
1 3 4
2 1 2
```

Saída

```
1 Valores invalidos para a multiplicacao.
```

Bom trabalho! :)