

2º Exercício: Produto de Matrizes

Pela propriedade associativa, o produto de matrizes pode ser calculado de várias formas.

No caso de três matrizes, podemos fazer $(A_1 \cdot A_2) \cdot A_3$ ou então $A_1 \cdot (A_2 \cdot A_3)$, obtendo-se o mesmo resultado.

Entretanto, a ordem em que tais operações são realizadas pode ter grande impacto no tempo total de cálculo, pois, dependendo das dimensões da matriz, realizaremos uma maior ou menor quantidade de operações escalares (multiplicações de números *float*, por exemplo).

Sejam N matrizes A_1, A_2, \dots, A_N .

Suponha que estejam em uma sequência tal que sejam compatíveis para multiplicação.

Isto é, o número de colunas de uma matriz é igual ao número de linhas da matriz seguinte.

Queremos descobrir a melhor ordem para efetuar as multiplicações.

Existem algoritmos eficientes para resolver esse problema.

Neste exercício, porém, nosso objetivo é utilizar recursividade.

Item (a)

Faça um programa **recursivo** para receber as dimensões das N matrizes a serem multiplicadas na sequência dada e calcular o número mínimo necessário de multiplicações escalares para efetuar esse produto de matrizes.

No final diga quantas chamadas (à função recursiva) foram realizadas durante a execução.

Item (b)

Mostrar qual é essa melhor ordem em que os produtos devem ser efetuados.

Observações:

Arquivo de entrada: `entrada2.txt`

Arquivo de saída: `Lab2_seu_nome_e_sobrenomes.txt`

$2 \leq N \leq 20$

A entrada terá o formato do exemplo.

A saída do item (b) deve seguir **exatamente** o formato do exemplo.