

Lista 4

- 1) Crie um algoritmo que receba uma matriz e imprima sua diagonal principal.
- 2) Crie um algoritmo que receba uma matriz e imprima os números de sua diagonal principal caso a soma desses números seja maior do que 10.
- 3) Crie um algoritmo que receba uma matriz e imprima os números de sua diagonal principal caso não haja números menores do que 9 na diagonal principal.
- 4) Crie um algoritmo que receba uma matriz e imprima os números de sua diagonal principal caso todos os números da diagonal principal sejam maiores do que 5.
- 5) Crie um algoritmo que receba duas matrizes, uma de inteiros e outra de caracteres, e retorne a concatenação das duas, elemento por elemento (use `concat()`).
- 6) Crie um algoritmo que receba duas matrizes, A e B, e retorne o produto destas matrizes.
- 7) Crie um algoritmo que crie uma matriz L por C e retorne a sequência de Fibonacci armazenada por colunas na matriz (1,1,2,3,5,8...).
- 8) Crie um algoritmo que receba três inteiros (usuário digita cada um) e cria três matrizes (**X**, **Y**, e **Z**). O primeiro inteiro é a quantidade de linhas das matrizes, e o segundo é a quantidade de colunas. O terceiro inteiro (chamamos de *k*) é um valor usado para calcular os valores das matrizes **X**, **Y**, e **Z**. Esse algoritmo deve preencher a matriz **X** com o quadrado de *k* acrescido de 1 por célula da matriz (variando por coluna). A matriz **Y** deve ser preenchida com *k* somado com as respectivas células da matriz **X**. Finalmente matriz **Z** deve ter suas linhas preenchidas com o somatório das linhas da matriz **X**.

Exemplo. Entrada: linha=2, coluna=3, k= 2

X=	Y=	Z=																		
<table border="1"><tr><td>3</td><td>4</td><td>5</td></tr><tr><td>6</td><td>7</td><td>8</td></tr></table>	3	4	5	6	7	8	<table border="1"><tr><td>5</td><td>6</td><td>7</td></tr><tr><td>8</td><td>9</td><td>10</td></tr></table>	5	6	7	8	9	10	<table border="1"><tr><td>12</td><td>12</td><td>12</td></tr><tr><td>21</td><td>21</td><td>21</td></tr></table>	12	12	12	21	21	21
3	4	5																		
6	7	8																		
5	6	7																		
8	9	10																		
12	12	12																		
21	21	21																		