

Construa um TAD matriz numérica com as seguintes características:

- a) seja implementado utilizando dicionário(s).
- b) elementos iguais a zero não devem ser armazenados.

Funções do TAD:

- c) `criar(linhas, colunas)`: cria e retorna um TAD matriz de dimensões linhas, colunas dadas. Retorna `None` se os valores de dimensões forem inválidas.
- d) `igual(TADMtrizA, TADMtrizB)`: retorna `True` se os TADMtrizes passados como argumento são iguais, `False` caso contrário.
- e) `determinante(TADMtrizA)`: retorna o valor do determinante do TADMtriz passado como parâmetro. Retorna `None` se o parâmetro não suportar determinante.
- f) `inversa(TADMtrizA)`: pesquise e implemente um algoritmo para o cálculo da inversa de uma matriz.
- g) `similarCos(TADMtrizA, TADMtrizB)`: implemente o método do cosseno (pesquise) para avaliar o nível de similaridade entre dois tadmtrizes.
- h) `similarJac(TADMtrizA, TADMtrizB)`: implemente o método de Jaccard (pesquise) para avaliar o nível de similaridade entre dois tadmtrizes.
- i) `vizinhos(TADMtrizA, i, j)`: retorna os 8 vizinhos de uma determinada posição `i, j` do tadmtriz de entrada. Vizinhos que não existem devem receber valor `None`. Ver a explicação do professor.
- j) `transposta(TADMtrizA)`: retorna a matriz transposta do parâmetro de entrada.
- k) `similar(TADMtrizA, TADMtrizB)`: pesquise e implemente um terceiro método de avaliação de similaridade entre os tadmtrizes de entrada.
- l) `multip(TADMtrizA, TADMtrizB)`: retorna a matriz produto das matrizes de entrada (tads). Retorna `None` se as matrizes não puderem ser multiplicadas.
- m) `andMat(TADMtrizA, TADMtrizB)`: retorna a matriz AND das matrizes de entrada. Cada elemento dessa matriz é o resultante and lógico dos elementos correspondentes dos tadmtrizes de entrada. Faça o mesmo para as operações lógicas de NOT, OR e XOR (pesquise as funções/operadores python que executam essas operações.).
- o) `adElem(TADMtriz, elem, i, j)`: altera a matriz de entrada adicionando o elemento na posição `i, j` do tadmtriz. Caso `i, j` sejam posições inválidas, retornar `None`. Retornar a matriz em caso de sucesso.
- p) `getElem(TADMtriz, i, j)`: retorna o elemento armazenado na posição `i, j` de TADMtriz.
- q) `extraí(TADMtriz, i, j, alt, larg)`: extraí uma submatriz (tadmtriz) de dimensões `alt, larg` a partir da posição `i, j` de TADMtriz. Retorna `None` se a submatriz (tad) não puder ser retornado.
- r) `insere(TADMtrizA, TADMtrizB, i, j)`: insere a matriz B na amtriz A a aprtir da posição `i, j`. Retorna `None` se a matriz B não puder ser inserida.