Instituto Federal de Educação, Ciência e Tecnologia de Brasília – Câmpus Taguatinga
Ciência da Computação – Programação de Computadores I
Lista de Exercícios – Matrizes
Prof. Daniel Saad Nogueira Nunes

Aluno: ____
Matrícula: ____

Exercício 1

Faça um programa que leia duas matrizes de números reais $A_{n\times m}$ $(1 \leq n, m \leq 100)$ e $B_{k\times l}$ $(1 \leq k, l \leq 100)$ e compute o produto matricial em uma matriz C. Seu programa deverá checar as condições de existência do produto matricial.

Exercício 2

Faça um programa que leia uma matriz $A_{n\times m}$ $(1 \leq n, m \leq 100)$ e compute $A^T_{m\times n}$.

Exercício 3

Em uma matriz quadrada, a k-ésima diagonal corresponde aos elementos da forma $a_{i,j}$ com $j = i + k$. Faça um programa que leia uma matriz quadrada de números reais $A_{n\times n}$ $(1 \leq n \leq 100)$ e diga qual a diagonal que possui a maior soma. Note que k está no intervalo $[-n+1, n-1]$, por exemplo, para uma matriz $A_{4\times 4}$, temos a presença das diagonais $\{-3, -2, -1, 0, 1, 2, 3\}$.

Exercício 4

Suponha que tenhamos n cidades, identificadas por números de 0 a $n-1$. A ligação entre a cidade i e a cidade j é dada pelo elemento $M[i][j]$ de uma matriz, isto é, $M[i][j] = 1$ se existe uma via da cidade i à cidade j. Note que se existe uma via de i a j, não necessariamente existe uma via de j a i, somente se $M[j][i]$ também possuir o valor 1.

Por exemplo, considere a seguinte matriz de adjacências:

Tabela 1: Matriz de Adjacências

	0	1	2	3
0	0	1	0	1
1	0	0	0	1
2	0	1	0	0
3	1	1	1	0

Neste exemplo, a cidade 0 possui uma estrada para as cidades 1 e 3. A cidade 1 possui uma estrada para a cidade 3 apenas. A cidade 2 possui estrada para a cidade 1 e a cidade 3 possui estrada para todas as outras.

Faça um programa que leia uma matriz quadrada $A_{n\times m}$ $(0 \leq n, m \leq 100)$ de valores booleanos (0 ou 1), elabore funções que:

(a) Dado um k lido do usuário, determine quantas estradas saem e quantas chegam à cidade k.

(b) A qual das cidades chega o maior número de estradas?

(c) Dado k lido do usuário, verificar se todas as ligações entre a cidade k e as demais são de mão dupla.

(d) Listar as cidades que possuem saídas para a cidade k lida do usuário.

(e) Listar as cidades isoladas, que não possuem ligações com nenhuma outra.

(f) Listar cidades que não possuem saídas (apesar de poderem possuir entradas).

(g) Listar cidades que há entradas, mas não saídas.

Exercício 5

Elabore um programa que leia uma matriz de inteiros $A_{n\times m}$ $(1 \leq n, m \leq 100)$ e um parâmetro $k(1 \leq k \leq \min\{n, m\})$ e imprima a posição da primeira célula da submatriz $B_{k\times k}$ de maior soma dentro de A. Além de imprimir a posição de B em relação a A, o seu programa também deverá imprimir a soma dos elementos de B.

Por exemplo, se a Matriz A corresponde à Tabela 2 e o parâmtro $k = 2$, a submatriz quadrada 2×2 de maior soma começa na posição $(1, 1)$ de A e tem soma 33. É a submatriz que possui os elementos $(7, 1, 10, 15)$.

Tabela 2: Matriz A.

5	2	12
3	7	1
9	10	15

Exercício 6

Leia uma matriz quadrada de reais $A_{n\times n}$ $(1 \leq n \leq 3)$ e compute o seu determinante.

Exercício 7

Um quadrado mágico é uma matriz quadrada $A_{n\times n}$ em que os números inteiros contidos nela são todos distintos e no intervalo $[1, n^2]$. Além disso, a soma de qualquer linha, coluna ou diagonal é igual, como na Figura 7.

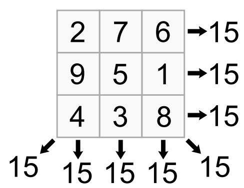

Figura 1: Quadrado mágico de ordem 3.

A ordem de um quadrado mágico corresponde ao número de linhas (que também é igual ao número de colunas).

(a) Faça um programa que leia um inteiro n ($1 \leq n \leq 100$) e uma matriz $A_{n \times n}$ e diga se aquela matriz corresponde a um quadrado mágico ou não.

(b) Faça um programa que leia um inteiro n ($1 \leq n \leq 100$) **ímpar** e faça um programa que preencha o quadrado mágico.

Dica: `https://en.wikipedia.org/wiki/Magic_square`.

Exercício 8

Leia uma matriz de reais $A_{n \times m}$ ($1 \leq n, m \leq 100$) e troque a coluna 0 com a coluna $m - 1$, a coluna 1 com a coluna $m - 2$ e assim sucessivamente. Imprima a matriz resultado.

Exercício 9

Leia um inteiro n ($1 \leq n \leq 100$) em seguida, leia n nomes com até 80 caracteres. Imprima cada um dos nomes na ordem inversa ao que foram lidos.

Exercício 10

Leia um inteiro n ($1 \leq n \leq 100$) em seguida, leia n nomes com até 80 caracteres. Diga se algum nome corresponde ao inverso de outro e indique quais são estes nomes.

Exercício 11

Leia um inteiro n ($1 \leq n \leq 100$) em seguida, leia n nomes com até 80 caracteres. Diga qual foi o nome que mais apareceu e a sua quantidade de vezes. Em caso de empate, o programa deverá dar prioridade para o nome que apareceu primeiramente na lista.

Exercício 12

Leia um inteiro n ($1 \leq n \leq 100$) em seguida, leia n nomes com até 80 caracteres. Por fim, leia um nome str. Diga se o nome str encontra-se na lista de nomes lidos.

Exercício 13

(**Jogo da vida de Conway**) O jogo da vida é um autômato celular inventado pelo matemático John Conway. Ele procura reproduzir através de regras simples, as alterações e mudanças em grupos de seres vivos, tendo aplicações em diversas áreas da ciência.

As regras são aplicadas a cada nova geração em um tabuleiro bi-dimensional definido pelo jogador. Dependendo do padrão inicial presente no tabuleiro, percebem-se mudanças muitas das vezes inesperadas e belas a cada nova geração.

As regras do jogo da vida, aplicadas a cada geração, são as seguintes:

1. Qualquer célula viva com menos de dois vizinhos vivos morre de solidão.
2. Qualquer célula viva com mais de três vizinhos morre de superpopulação.
3. Qualquer célula morta com exatamente três vizinhos torna-se uma célula viva.
4. Qualquer célula viva com dois ou três vizinhos continua no mesmo estado para a próxima geração.

Todos os nascimentos e mortes dentro de uma mesma geração ocorrem simultaneamente. O conceito de vizinho empregado engloba as diagonais.

Faça um programa que leia uma matriz $A_{n \times m}$ $(1 \leq n, m \leq 100)$ contendo espaços vazios e '*', representando células, e um parâmetro k $(1 \leq k \leq 100)$ e simule A por k gerações seguindo as regras do jogo. Seu programa deve imprimir o resultado de cada simulação.

Alguns padrões podem ter efeitos bem legais: `https://en.wikipedia.org/wiki/Conway%27s_Game_of_Life#Examples_of_patterns`