

## Fundamentos da programação

### lista extra 05: praticando a programação com matrizes

#### 1. Celebridade

Imagine que  $C$  é uma matriz que contém apenas os números 0 e 1.

A ideia é que a matriz  $C$  representa a seguinte relação entre pessoas

- se  $C[i][j] = 1$ , então a pessoa  $i$  conhece a pessoa  $j$
- se  $C[i][j] = 0$ , então a pessoa  $i$  não conhece a pessoa  $j$

A *celebridade* é a pessoa que é conhecida pelo maior número de outras pessoas.

Faça um programa que descobre quem é a celebridade.

#### 2. Ordenação por colunas

Considere a seguinte regra para comparar as colunas de uma matriz

- Se o maior elemento da coluna  $C_i$  é maior que o maior número da coluna  $C_j$  então nós dizemos que  $C_i$  é maior que  $C_j$

Faça um programa que ordena as colunas de uma matriz de acordo com essa regra

— (i.e., as colunas menores devem ficar à esquerda, e as colunas maiores devem ficar à direita)

#### 3. Diagonais

a) Imprima as diagonais de uma matriz com dimensão  $n \times n$ .

Por exemplo, para  $n = 4$ , nós teríamos o seguinte

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} \Rightarrow \begin{array}{l} 13 \\ 9, 14 \\ 5, 10, 15 \\ 1, 6, 11, 16 \\ 2, 7, 12 \\ 3, 8 \\ 4 \end{array}$$

b) Imprima as diagonais opostas de uma matriz com dimensão  $n \times n$ .

Por exemplo, para  $n = 4$ , nós teríamos o seguinte

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} \Rightarrow \begin{array}{l} 1 \\ 5, 2 \\ 9, 6, 3 \\ 13, 10, 7, 4 \\ 14, 11, 8 \\ 15, 12 \\ 16 \end{array}$$

#### 4. Sudoku (Difícil)

Imagine que as posições de uma matriz **S** com dimensão  $9 \times 9$  correspondem às casas de um jogo de Sudoku.

Algumas dessas posições já estão preenchidas com números de 1 a 9, e outras casas estão vazias (o que é representado na matriz com o número 0).

Faça um programa que resolve o jogo de Sudoku.

**Dica:** Utilize uma matriz auxiliar que contém uma lista para cada casa do diagrama

```
Aux [9] [9] [9] ;
```

A ideia é que cada lista na matriz **Aux** indica os valores que ainda estão disponíveis para a casa correspondente.

E utilizando essa informação, você pode aplicar os seguintes passos de raciocínio

- Sempre que uma lista contém apenas um elemento, a casa correspondente já pode ser preenchida.
- E sempre que você preenche uma casa, você deve excluir aquele número das listas de outras casas (de acordo com as regras do Sudoku).

Repita esses passos sucessivamente até que o diagrama esteja completo

— (*ou então indique que o jogo não pode ser resolvido dessa maneira*)

bla