

Lista_05

October 23, 2020

1 Lista de Exercícios 5

```
[1]: import numpy as np
import pandas as pd
```

1.1 QUESTÃO 1:

Implementar a ordenação de matrizes pelo método do [Bubble Sort](#). A função deve receber dois parâmetros de entrada: (i) uma matriz e (ii) o tipo de ordenamento (linha ou coluna).

```
[ ]:
```

1.2 QUESTÃO 2:

Criar uma função que recebe uma matriz como parâmetro de entrada e troca 2 linhas (ou colunas) escolhidas (parâmetros de entrada) de posição. A escolha da troca por 'linha' ou 'coluna' também é um parâmetro de entrada.

```
[ ]:
```

1.3 QUESTÃO 3:

Criar uma função que recebe uma matriz como parâmetro de entrada e inverte a ordem das linhas (ou colunas) dessa matriz.

```
[ ]:
```

1.4 QUESTÃO 4:

Criar uma função para multiplicar duas matrizes, pela implementação do método de multiplicação matricial (não usar a função de multiplicação de matrizes do Numpy).

```
[ ]:
```

1.5 QUESTÃO 5:

Calcular o [determinante](#) de uma matriz usando o **método de escalonamento**.

[]:

1.6 QUESTÃO 6:

Calcular o [determinante](#) de uma matriz usando o **método dos cofatores** (recursivo).

[]:

1.7 QUESTÃO 7:

Contar os pontos do boliche, as regras e um exemplo de jogo bem detalhado estão no arquivo `boliche-regras-de-pontos.docx`. Use os seguintes dados:

```
[2]: #Dados: boliche
jog1=[[10,0],[10,0],[10,0],[7,2],[8,2],[0,9],[10,0],[7,3],[9,0],[10,0],[10,8]]
jog2=[[2,5],[5,5],[10,0],[6,3],[10,0],[10,0],[9,0],[5,2],[9,1],[9,0]]
jog3=[[9,1],[10,0],[8,1],[5,4],[10,0],[10,0],[10,0],[9,1],[8,7],[8,2],[5,0]]
jog4=[[2,2],[4,1],[8,1],[6,2],[9,1],[0,10],[1,5],[2,2],[4,2],[4,1]]
```

[]:

1.8 QUESTÃO 8:

Campo minado: Defina o tamanho n do quadro do jogo e a quantidade p de bombas no quadro. Crie uma função que recebe como parâmetro de entrada uma matriz $p \times 2$, com a lista das posições onde estão localizadas as bombas (em cada linha, a primeira coluna contém o índice da linha e a segunda coluna o índice da coluna onde está localizada a bomba); e retorna como saída a matriz de vizinhança, de dimensão $n \times n$, contendo, em cada célula, a quantidade de bombas existentes nas células vizinhas.

As células onde estão localizadas as bombas devem ser preenchidas com NaN (not a number - `np.nan`) e as células que não têm bombas nas vizinhanças devem ser preenchidas com zeros.

[]:

1.9 QUESTÃO 9:

Uma das tarefas computacionais importantes é a localização de padrões de repetições em conjuntos de dados. Sua tarefa nesta questão é desenvolver um código para localizar e contar as ocorrências de uma sequência de valores dentro de uma matriz.

- Criar uma função que recebe como parâmetros de entrada (i) uma matriz de dados; e (ii) um valor a ser procurado, e retorna como parâmetros de saída (i) a quantidade de vezes que este valor aparece na matriz, e (ii) uma lista com as posições (linha coluna) onde esses valores são encontrados.

```
[3]: dados=np.matrix(pd.read_excel('ocorrencias.xlsx',header=None))
seq1=np.matrix(pd.read_excel('ocorrencias.xlsx',header=None,sheet_name="seq1"))
seq2=np.matrix(pd.read_excel('ocorrencias.xlsx',header=None,sheet_name="seq2"))
```

```
seq3=np.matrix(pd.read_excel('ocorrencias.xlsx',header=None,sheet_name="seq3"))
```

[]:

- b) Criar uma função que recebe como parâmetros de entrada uma matriz de dados e uma sequência de valores (array ou lista), e localiza a sequência na matriz, retornando a lista com as posições (i, j) onde a sequência foi encontrada.

Por exemplo, teremos o seguinte resultado quando procuramos a sequência [1 2 3] na matriz abaixo:

8	10	14	13	3
2	1	2	3	14
3	6	2	11	11
5	12	1	2	3
7	15	4	9	3
1	2	3	4	7

N.Rep: 3

1	1
3	2
5	0

Você pode testar sua função com os dados disponíveis na planilha dados e com os vetores seq1, seq2 e seq3, do arquivo `ocorrencias.xls`. Os resultados esperados são os seguintes:

Seq1

N.Rep: 3

14	3
66	25
115	45

Seq2

N.Rep: 5

4	19
19	2
20	37
281	34
439	16

Seq3

N.Rep: 2

10	40
36	4

[]:

1.10 QUESTÃO 10:

Troca-troca de figurinhas: Criar uma função que recebe dois pares de listas, representando as figurinhas de dois jogadores: para cada jogador, passe a lista de figurinhas que ele precisa e a lista de figurinhas que ele tem duplicatas. Retorne com a lista de trocas que esses dois jogadores podem fazer entre si.

[]: