**MAC 122 - PDA**

**Segundo Semestre de 2020 – BMAC – IMEUSP**

**Exercícios II – Entregar até 15/Outubro/2020**

**Algoritmos Recursivos - Sudoku**

A função Sudoku abaixo, encontra todas as soluções a partir de uma matriz dada.

Completar a função com os comandos que faltam:

**# Preenche a matriz Sudoku**

**def Sudoku(S, L, C):**

 **Procure uma posição livre a partir de S[L][C]**

 **Se não há:**

 **# A matriz S está completa.**

 **Mostre S.**

 **retorne desta chamada**

 **Seja S[lin][col] a posição livre.**

 **# Podem haver zero ou mais candidatos a ocupar essa posição.**

 **# Os candidatos não podem estar na linha lin, na coluna col**

 **# ou no quadrado correspondente.**

 **Para cada candidato:**

 **S[lin][col] = candidato.**

 **Sudoku(S, lin, col)**

 **# Neste ponto não há candidato para a posição S[lin][col]**

 **# Libera e retorna para que chamadas anteriores da função**

 **# possam tentar o preenchimento**

 **S[lin][col] = 0**

 **retorne desta chamada**

Teste sua função com o programa abaixo que lê arquivos contendo uma matriz incompleta e preenche a matriz com as soluções possíveis.

A matriz sudoku possui 9 linhas e 9 colunas. Posições com zero são casas vazias a serem preenchidas.

**from sudoku import \***

**def LeiaMatriz(Mat, NomeArq):**

 **# retorna True se conseguiu ler o arquivo e False caso contrário**

 **# abrir o arquivo para leitura**

 **try:**

 **arq = open(NomeArq, "r")**

 **except:**

 **return False**

 **# ler cada uma das linhas**

 **i = 0**

 **for linha in arq:**

 **v = linha.split()**

 **# transforma os strings em números**

 **for j in range(len(v)):**

 **Mat[i][j] = int(v[j])**

 **i = i + 1**

 **# fechar arquivo**

 **arq.close()**

 **return True**

**def MostreMatriz(Mat, tit):**

 **print("\n", tit)**

 **for i in range(9):**

 **print()**

 **for j in range(9):**

 **print(Mat[i][j], end = ' ')**

 **print()**

**# Lê arquivos contendo uma matriz sudoku**

**# Para cada arquivo lido chama a função sudoku**

**# Até que seja digitado 'fim'**

**def main():**

 **# inicia matriz sudoku**

 **mats = [9 \* [0] for i in range(9)]**

 **# repita até ser digitado fim como nome do arquivo**

 **while True:**

 **# nome do arquivo**

 **while True:**

 **nomearq = input("\n\nEntre com o nome do arquivo:")**

 **if nomearq == "fim": return**

 **# ler e mostrar a matriz inicial**

 **if LeiaMatriz(mats, nomearq): break**

 **else: print("\* \* \* Arquivo inválido \* \* \*")**

 **MostreMatriz(mats, "\n\n\* \* \* Matriz inicial \* \* \*")**

 **# Chamada da função Sudoku**

 **# Encontra todas as soluções para esta matriz**

 **Sudoku(mats, 0, 0)**

**# executa o main()**

**main()**

Serão fornecidos também vários arquivos para o seu teste com nomes:

**sudokuteste1.txt**

**sudokuteste2.txt**

**...**

**sudokuteste10.txt**

Entregue apenas o módulo sudoku.py contendo a sua função Sudoku e eventualmente as funções auxiliares que precisar. Note que no início do programa de teste há:

**from sudoku import \***