

Sudoku Solver und Generator

Idee des Solvers

Das Sudoku muss in Nested List vorliegen

Algorithmus:

1. den nächsten leeren Platz im Sudoku finden (also mit Eintrag 0)
2. Eintragen einer Zahl von 1-9 in diesem Feld
3. Check, ob die eingetragene Zahl möglich ist (also unique in Zeile, Spalte und jeweiligem 3x3 Quadrat ist)
4. Falls Valid, rekursives Füllen der anderen Felder, wenn nicht, eintragen von 0 in die aktuelle Stelle und Wiederholung des vorigen Schrittes
5. Wenn keine Nullen mehr vorhanden sind, ist das Sudoku gelöst

Idee des Generators

Algorithmus:

1. Erzeugen eines zufälligen Sudokus mit Hilfe des Solvers aus einem Anfangs Sudoku mit nur Nullen, deshalb muss im Solver die num_list zufällig gewählt werden,
sonst erhalten wir immer das selbe Sudoku
2. Auswählen einer zufälligen Stelle und 0 setzen dieser
3. checken ob bei zurücksetzen Feld eine eindeutige Lösung vorhanden ist, wenn ja, Wiederholung des Prozessen wenn nein, zurücksetzen des Feldes auf ursprüngliche Zahl
4. Abbruchkriterium, wenn gewünschte Anzahl an leeren Feldern erzeugt wurden oder beim 10. Versuch kein neues Feld mit eindeutiger Lösung gefunden wird

Wie wird das Programm genutzt?

Solver:

Aufruf des Programms in der Kommandozeile mit Eingabe der Datei, welche das Sudoku in csv Form beinhaltet

z.B. mit Eingabe --> `python solve_sudoku.py sudoku_1.csv`

Generator:

Aufruf des Programms in der Kommandozeile mit Eingabe der gewünschten leeren Feldanzahl

z.B. mit Eingabe --> `python sudoku_generator.py 15`

Quellen

<https://www.techwithtim.net/tutorials/python-programming/sudoku-solver-backtracking/>

<https://www.askpython.com/python/examples/sudoku-solver-in-python>

<https://medium.com/@ev.zafeiratos/sudoku-solver-with-python-a-methodical-approach-for-algorithm-optimization-part-1-b2c99887167f>