Articles › 36. Valid Sudoku

Previous (/articles/search-in-rotated-sorted-array/) Next (/articles/odd-even-jump/)

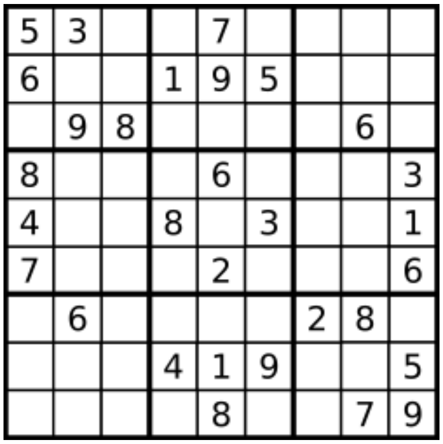
36. Valid Sudoku (/problems/valid-sudoku/)

Jan. 11, 2019 | 15.4K views

Average Rating: 3.20 (30 votes)

Determine if a 9x9 Sudoku board is valid. Only the filled cells need to be validated **according to the following rules**:

1. Each row must contain the digits `1-9` without repetition.
2. Each column must contain the digits `1-9` without repetition.
3. Each of the 9 `3x3` sub-boxes of the grid must contain the digits `1-9` without repetition.



5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

A partially filled sudoku which is valid.

The Sudoku board could be partially filled, where empty cells are filled with the character `'.'`.

Example 1:

```
Input:
[
  ["5","3",".",".","7",".",".",".","."],
  ["6",".",".","1","9","5",".",".","."],
  [".","9","8",".",".",".",".","6","."],
  ["8",".",".",".","6",".",".",".","3"],
  ["4",".",".","8",".","3",".",".","1"],
  ["7",".",".",".","2",".",".",".","6"],
  [".","6",".",".",".",".","2","8","."],
  [".",".",".","4","1","9",".",".","5"],
  [".",".",".",".","8",".",".","7","9"]
]
Output: true
```

Example 2:

```
Input:
[
  ["8","3",".",".","7",".",".",".","."],
  ["6",".",".","1","9","5",".",".","."],
  [".","9","8",".",".",".",".","6","."],
  ["8",".",".",".","6",".",".",".","3"],
  ["4",".",".","8",".","3",".",".","1"],
  ["7",".",".",".","2",".",".",".","6"],
  [".","6",".",".",".",".","2","8","."],
  [".",".",".","4","1","9",".",".","5"],
  [".",".",".",".","8",".",".","7","9"]
]
Output: false
Explanation: Same as Example 1, except with the 5 in the top left corner being
    modified to 8. Since there are two 8's in the top left 3x3 sub-box, it is invalid.
```

Note:

- A Sudoku board (partially filled) could be valid but is not necessarily solvable.
- Only the filled cells need to be validated according to the mentioned rules.
- The given board contain only digits `1-9` and the character `'.'`.
- The given board size is always `9x9`.

Solution

Intuition

The naive solution would be to iterate *three* times over the board to ensure that :

- There is no rows with duplicates.
- There is no columns with duplicates.
- There is no sub-boxes with duplicates.

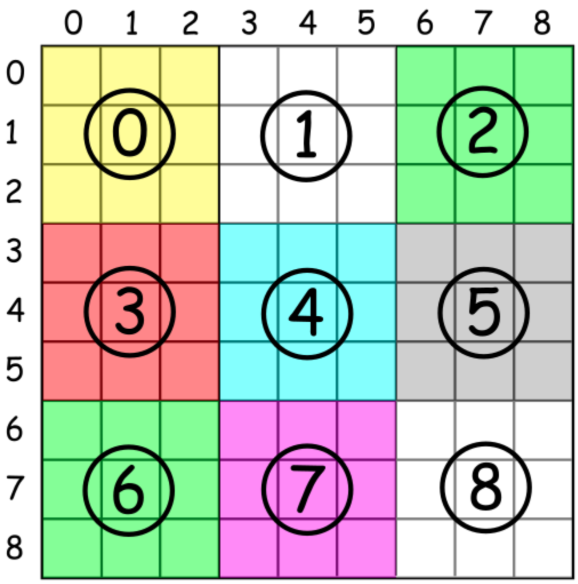
Actually, all this could be done in just one iteration.

Approach 1: One iteration

Let's first discuss two questions.

- How to enumerate sub-boxes?

> One could use `box_index = (row / 3) * 3 + col / 3` where `/` is an integer division, `row` is a row number, and `col` is a column number.



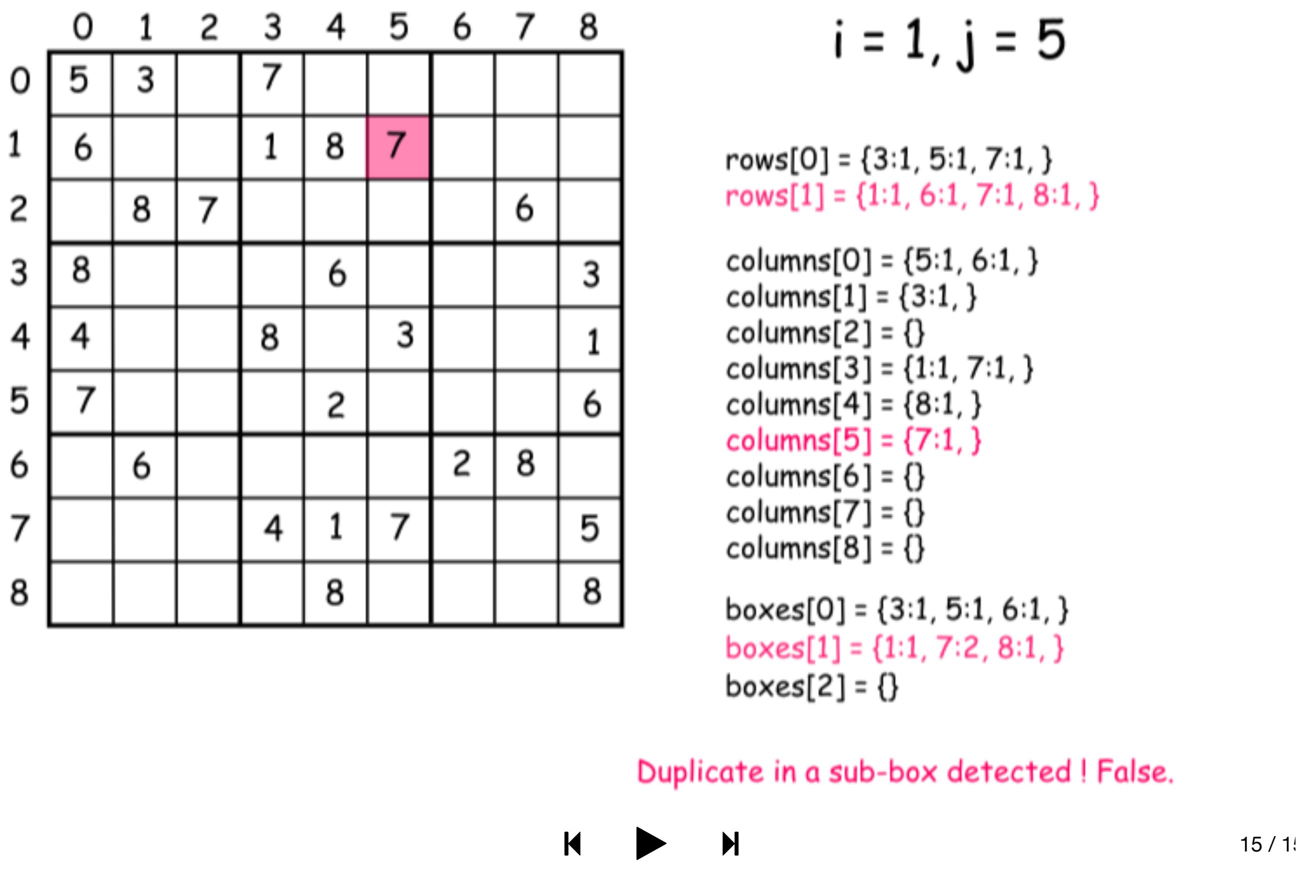
- How to ensure that there is no duplicates in a row / column / box?

> One could just track all values which were already encountered in a hash map `value -> count`.

Now everything is ready for the overall algorithm :

- Move along the board.
- Check for each cell value if it was seen already in the current row / column / box :

  - Return `false` if yes.
  - Keep this value for a further tracking if no.
- Return `true`.



	0	1	2	3	4	5	6	7	8
0	5	3		7					
1	6			1	8	7			
2		8	7					6	
3	8				6				3
4	4			8		3			1
5	7				2				6
6		6					2	8	
7				4	1	7			5
8					8				8

i = 1, j = 5

rows[0] = {3:1, 5:1, 7:1, }
rows[1] = {1:1, 6:1, 7:1, 8:1, }

columns[0] = {5:1, 6:1, }
columns[1] = {3:1, }
columns[2] = {}
columns[3] = {1:1, 7:1, }
columns[4] = {8:1, }
columns[5] = {7:1, }
columns[6] = {}
columns[7] = {}
columns[8] = {}

boxes[0] = {3:1, 5:1, 6:1, }
boxes[1] = {1:1, 7:2, 8:1, }
boxes[2] = {}

Duplicate in a sub-box detected ! False.

Java Python Copy

```python
class Solution:
    def isValidSudoku(self, board):
        """
        :type board: List[List[str]]
        :rtype: bool
        """
        # init data
        rows = [{} for i in range(9)]
        columns = [{} for i in range(9)]
        boxes = [{} for i in range(9)]

        # validate a board
        for i in range(9):
            for j in range(9):
                num = board[i][j]
                if num != '.':
                    num = int(num)
                    box_index = (i // 3 ) * 3 + j // 3

                    # keep the current cell value
                    rows[i][num] = rows[i].get(num, 0) + 1
                    columns[j][num] = columns[j].get(num, 0) + 1
                    boxes[box_index][num] = boxes[box_index].get(num, 0) + 1

                    # check if this value has been already seen before
                    if rows[i][num] > 1 or columns[j][num] > 1 or boxes[box_index][num] > 1:
                        return False
```

Complexity Analysis

- Time complexity : $\mathcal{O}(1)$ since all we do here is just one iteration over the board with 81 cells.
- Space complexity : $\mathcal{O}(1)$.

Analysis written by @liaison (https://leetcode.com/liaison/) and @andvary (https://leetcode.com/andvary/)

Rate this article:

Previous (/articles/search-in-rotated-sorted-array/)

Next (/articles/odd-even-jump/)

Comments: 16

Sort By

Type comment here... (Markdown is supported)

Preview Post

(/zli_test)

zli_test (zli_test) ★ 51 January 16, 2019 11:38 AM

This solution is a typical example for smart coding, which is pretty bad. One iteration does 3 things vs 3 iterations each does one thing. There is no difference in time complexity, and space complexity. As a software engineer, you should write clean code which is easy to understand.

43 ^ v Share Reply

SHOW 3 REPLIES

(/chuyao)

chuyao (chuyao) ★ 11 June 1, 2019 7:17 PM

Why not use HashSet...

9 ^ v Share Reply

SHOW 1 REPLY

(/indish)

indish (indish) ★ 69 April 14, 2019 9:45 PM

How to come up with this equation? trial and error? any systematic approach?

```
box_index = (row / 3) * 3 + col / 3
```

Too much math, wrote the below helper function to get box index.

Read More

4 ^ v Share Reply

SHOW 1 REPLY

(/cyrusmith)

cyrusmith (cyrusmith) ★ 37 January 29, 2019 7:20 AM

This problem should be qualified as easy, not medium.

6 ^ v Share Reply

(/algorithmimplementer)

AlgorithmImplementer (algorithmimplementer) ★ 31 August 2, 2019 2:59 PM

Why is the space O(1)? I can see that 3 hash maps are created as auxiliary storage.

1 ^ v Share Reply

SHOW 1 REPLY

(/abiaps)

abiaps (abiaps) ★ 30 January 24, 2019 7:51 PM

Can anyone explain how the box index is calculated please?

1 ^ v Share Reply

SHOW 3 REPLIES

(/devkapupara)

devkapupara (devkapupara) ★ 32 January 12, 2019 6:06 PM

Wait, how is it constant space? As for the time complexity, is it constant time just becuase the number of iterations are known? Why not O(n^2)?

1 ⌃ ⌄ Share Reply

SHOW 3 REPLIES

(/binglux)

binglux (binglux) ★ 0 April 2, 2019 11:10 AM

Python3 52ms, beat 95.42%, first make combinations then see if the number of elements more than 1 in each combination.

```
class Solution:
    def isValidSudoku(self, board: List[List[str]]) -> bool:
```

Read More

0 ⌃ ⌄ Share Reply

(/berlet)

berlet (berlet) ★ 1 March 26, 2019 12:12 PM

I'm a little confused with the IO here (since I can't see the main method equivalent in the code). Why does the method header list a char grid if the input is a String grid that isn't in proper Java array-grid syntax(e.g. {{"9","2","3"},{"6",".","1"}})? Code that works in my IDE doesn't process correctly here...

0 ⌃ ⌄ Share Reply

(/root32)

root32 (root32) ★ 0 January 19, 2019 12:16 PM

Can this be fine ? it worked with the given example.

data = [
["5","3",".",".","7",".",".",".","."],
["6",".",".","1","9","5",".",".","."],

Read More

0 ⌃ ⌄ Share Reply

‹ 1 2 ›

Help Center (/support/) | Students (/students) | Terms (/terms/) | Privacy