36. Valid Sudoku

Created	@July 19, 2020 9:37 PM
Difficulty	Medium
LC Url	https://leetcode.com/problems/valid-sudoku/
Importance	***
Tag	Array&Sorting NEET
Video	https://www.youtube.com/watch?v=TjFXEUCMqI8&t=42s

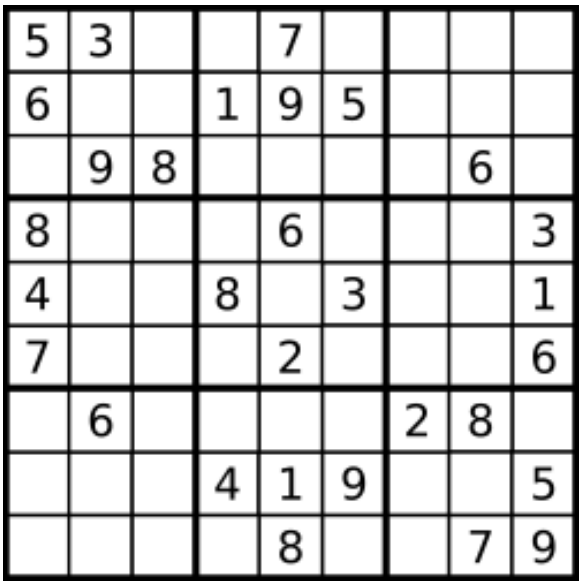
Determine if a `9 x 9` Sudoku board is valid. Only the filled cells need to be validated **according to the following rules**:

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 nine `3 x 3` sub-boxes of the grid must contain the digits `1-9` without repetition.

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.

Example 1:




```
Input: board =
[["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: board =
[["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 the5 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.
```

Constraints:

- `board.length == 9`
- `board[i].length == 9`
- `board[i][j]` is a digit `1-9` or `'.'`.

Solution

方法：一次遍历

有效的数独满足以下三个条件：

- 同一个数字在每一行只能出现一次；
- 同一个数字在每一列只能出现一次；
- 同一个数字在每一个小九宫格只能出现一次。

可以使用哈希表记录每一行、每一列和每一个小九宫格中，每个数字出现的次数。只需要遍历数独一次，在遍历的过程中更新哈希表中的计数，并判断是否满足有效的数独的条件即可。

对于数独的第 i 行第 j 列的单元格，其中 $0 \leq i, j < 9$，该单元格所在的行下标和列下标分别为 i 和 j，该单元格所在的小九宫格的行数和列数分别为 $\left\lfloor \frac{i}{3} \right\rfloor$ 和 $\left\lfloor \frac{j}{3} \right\rfloor$，其中 $0 \leq \left\lfloor \frac{i}{3} \right\rfloor, \left\lfloor \frac{j}{3} \right\rfloor < 3$。

由于数独中的数字范围是 1 到 9，因此可以使用数组代替哈希表进行计数。

具体做法是，创建二维数组 $rows$ 和 $columns$ 分别记录数独的每一行和每一列中的每个数字的出现次数，创建三维数组 $subboxes$ 记录数独的每一个小九宫格中的每个数字的出现次数，其中 $rows[i][index]$、$columns[j][index]$ 和 $subboxes\left[\left\lfloor \frac{i}{3} \right\rfloor\right]\left[\left\lfloor \frac{j}{3} \right\rfloor\right][index]$ 分别表示数独的第 i 行第 j 列的单元格所在的行、列和小九宫格中，数字 $index+1$ 出现的次数，其中 $0 \leq index < 9$，对应的数字 $index+1$ 满足 $1 \leq index+1 \leq 9$。

如果 $board[i][j]$ 填入了数字 n，则将 $rows[i][n-1]$、$columns[j][n-1]$ 和 $subboxes\left[\left\lfloor \frac{i}{3} \right\rfloor\right]\left[\left\lfloor \frac{j}{3} \right\rfloor\right][n-1]$ 各加 1。如果更新后的计数大于 1，则不符合有效的数独的条件，返回 false。

如果遍历结束之后没有出现计数大于 1 的情况，则符合有效的数独的条件，返回 true。

```
class Solution:
    def isValidSudoku(self, board: List[List[str]]) -> bool:
        cols = collections.defaultdict(set)
        rows = collections.defaultdict(set)
        squares = collections.defaultdict(set)  # key = (r // 3, c // 3)
```

```
        for r in range(9):
            for c in range(9):
                if board[r][c] == ".":
                    continue
                if (
                    board[r][c] in rows[r]
                    or board[r][c] in cols[c]
                    or board[r][c] in squares[(r // 3, c // 3)]
                ):
                    return False
                cols[c].add(board[r][c])
                rows[r].add(board[r][c])
                squares[(r // 3, c // 3)].add(board[r][c])

        return True
```

复杂度分析

- 时间复杂度：$O(1)$。数独共有 81 个单元格，只需要对每个单元格遍历一次即可。
- 空间复杂度：$O(1)$。由于数独的大小固定，因此哈希表的空间也是固定的。

https://www.lintcode.com/problem/389/solution/57196