36. Valid Sudoku

Medium

696242FavoriteShare

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.



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.

class Solution {

public:

bool isValidSudoku(vector<vector<char>>& board) {

int used1[9][9]={0}, used2[9][9]={0}, used3[9][9]={0};

for(int i=0;i<9;i++){

for(int j=0;j<9;j++){

if(board[i][j]!='.'){

int num=board[i][j]-'0'-1;

int k=i/3\*3 + j/3;

if(used1[i][num]==1||used2[j][num]==1||used3[k][num]==1) return false;

used1[i][num]=used2[j][num]=used3[k][num]=1;

}

}

}

return true;

}

};

Success

[Details](https://leetcode.com/submissions/detail/205898273/)

Runtime: 8 ms, faster than 98.72% of C++ online submissions for Valid Sudoku.

Memory Usage: 774.1 KB, less than 90.44% of C++ online submissions forValid Sudoku.