

Solve Sudoku

1. You are given a partially filled 9*9 2-D array(arr) which represents an incomplete sudoku state.
2. You are required to assign the digits from 1 to 9 to the empty cells following some rules.

Rule 1 -> Digits from 1-9 must occur exactly once in each row.

Rule 2 -> Digits from 1-9 must occur exactly once in each column.

Rule 3 -> Digits from 1-9 must occur exactly once in each 3x3 sub-array of the given 2D array.

Assumption -> The given Sudoku puzzle will have a single unique solution.

Input Format

9*9 integers ranging from 1 to 9.

0 represents an empty cell.

Output Format

You have to print the solved sudoku.

Constraints

$0 \leq \text{arr}[i][j] \leq 9$

Sample Input

```
3 0 6 5 0 8 4 0 0
5 2 0 0 0 0 0 0 0
0 8 7 0 0 0 0 3 1
0 0 3 0 1 0 0 8 0
9 0 0 8 6 3 0 0 5
0 5 0 0 9 0 6 0 0
1 3 0 0 0 0 2 5 0
0 0 0 0 0 0 0 7 4
0 0 5 2 0 6 3 0 0
```

Sample Output

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

692351874
745286319

```
def sudoku(board):
    sudokuSolver(board, 0, 0)

def sudokuSolver(board, x, y):
    if x == len(board):
        print(board)
        return
    ni = 0
    nj = 0
    if y == len(board) - 1:
        ni = x + 1
        nj = 0
    else:
        ni = x
        nj = y + 1

    if board[x][y] != 0:
        sudokuSolver(board, ni, nj)
    else:
        for i in range(1, 10):
            if isValid(board, x, y, i):
                board[x][y] = i
                sudokuSolver(board, ni, nj)
                board[x][y] = 0

def isValid(board, x, y, val):
    for i in range(0, 9):
        if board[i][y] == val:
            return False

    for j in range(0, 9):
        if board[x][j] == val:
            return False

    ni = 3 * (x // 3)
    nj = 3 * (y // 3)

    for i in range(3):
```

```
        for j in range(3):
            if board[ni + i][nj + j] == val:
                return False

    return True
```

```
board = [[3, 0, 6, 5, 0, 8, 4, 0, 0],
          [5, 2, 0, 0, 0, 0, 0, 0, 0],
          [0, 8, 7, 0, 0, 0, 0, 3, 1],
          [0, 0, 3, 0, 1, 0, 0, 8, 0],
          [9, 0, 0, 8, 6, 3, 0, 0, 5],
          [0, 5, 0, 0, 9, 0, 6, 0, 0],
          [1, 3, 0, 0, 0, 0, 2, 5, 0],
          [0, 0, 0, 0, 0, 0, 0, 7, 4],
          [0, 0, 5, 2, 0, 6, 3, 0, 0]]
```

```
sudoku(board)
```