

## 1. Solve this puzzle using pseudo Code

//Checking if these conditions are true or false.

//Each of the digits 1-9 is exactly once in each row.

//Each of the digits 1-9 is exactly once in each column.

//Each of the digits 1-9 is exactly once in each of the 3x3 sub-boxes of the grid.

//Nested array of characters from 1 to 9 as a parameter (char[][] board).

```
bool isArrangementValid(char[][] board, int i, int j, char digit) {  
    for (int col = 0 to col < 9)  
        if (board[i][col] == digit)  
            return false;  
  
    for (int row = 0 to row < 9)  
        if (board[row][j] == digit)  
            return false  
  
    for (int row = (i / 3) * 3 to row < (i / 3 + 1) * 3)  
        for (int col = (j / 3) * 3 to col < (j / 3 + 1) * 3)  
            if (board[row][col] == digit)  
                return false  
  
    return true  
}
```

//Function to solve.

//Empty cells indicated by null.

```
bool solve(char[][] board) {  
    for (int i = 0 to i < 9) {  
        for (int j = 0 to j < 9) {  
            if (board[i][j] == null) {  
                for (char digit = '1'; digit <= '9'; digit++) {  
                    if (isValid(board, i, j, digit)) {  
                        board[i][j] = d  
                        if (solve(board))
```

```
        return true
        board[i][j] = null
    }
}
return false
}
}
}
return true
}
```

```
//Solving the puzzle using the board.
void solveSudokuPuzzle(char[][] board) {
    solve(board)
}
```