About Sudoku

Front End – reactjs with vanilla JS HTML & CSS

Back End – none

Hosted On – nginx server

Developer – Vivek sharma

Github –

Github

Twitter

Linkedin

Algorithm

**FillSudoku**

This program fills the 9\*9 grid with valid sudoku numbers. 2 dimension arrays play an important role in most 2d games. Sudoku is no exception, consider it as tiles which needs to fill up the screen at valid position. We pick a box try to insert a number there, run all the rules, if it works good if not try the next number.

Detailed Algorithm

Initialize all arrays used in the program

Create an array of 9 numbers – tmpjumbarr

Randomize tmpjumbarr

For each of the nine box do below

Pick first value from array tmpjumbarr and put it in the box

Check whether that number satisfied all the rules or not

* No same number in same row
* No same number in same column
* No same number in 3\*3 grid that box is part of

If the number satisfies fit it in the box, remove this number from tmpjumbarr and move on to next box

If the number fails, put this number at the end for another box and try the next number in the grid

Even after multiple tries we are not able to fill the grid, that means we are in deadlock, empty out the row and then start again

Even if we are still not able to fill the grid, deadlock condition initially we rollback couple of steps and then start again

**HideSudoku**

We are passing in difficulty level from frontend. Based on the difficulty chosen the program will decide how many columns to hide

Run below loop for the number of boxes to be hidden

Find a random i

Find a random j

Hide the array at position arr[i][j]

There could be possibility of repeating certain boxes, we are not so fussy about that. React uses these funtions to receive the 2d arrays filled and hidden and clones to run its check etc.