Shan Mehrotra
MT2014111

AIM :

To write a Sudoku Solver

INTRODUCTION :

SUDOKU is a logic-based number-placement puzzle
which is usually solved by hand for fun. For example, a 9X9 Sudoku puzzle may look as in
Fig. 1. The initial set of occu-pied cells are called the "clues" of the puzzle. The objective is
to fill the empty cells in the puzzle such that the digits 1....9 occur only once in each row,
each column and each 3X3 box.

Fig . 1 .

Taking the 9X9 Sudoku puzzle shown in Fig. 1 as an ex-ample, the four types of constraints
that one needs to satisfy to solve the puzzle are as follows.

• **Row constraints:** Each row of should comprise all digits

• **Column constraints:** Each column of should comprise all
digits

• **Box constraints:** Each 3 3 highlighted box of should
comprise all digits
.
• **Cell constraints:** Each cell of should be filled.

PROCEDURE :

The sudoku solver is implemented using Rule 1, Rule 2 and backtracking.

1. Rule 1 - This technique is very easy – especially if you're using pencilmarks to store what candidates are still possible within each cell.
If you've managed to rule out all other possibilities for a particular cell (by examining the surrounding column, row and box), so there's only one number left that could possibly fit there – you can fill in that number. In Fig .2. we can see the output given by Rule 1.

Fig . 2 .

2. Rule 2 – In this rule we have to scan all the nodes that contains multiple possible values and for a perticular node for a possible value we have to see whether that appears in a row or a column or a grid, if it does'nt appears in any one of these than we assign the value to that node.

3. Backtracking - Like all other Backtracking problems, we can solve Sudoku by one by one assigning numbers to empty cells. Before assigning a number, we check whether it is safe to assign. We basically check that the same number is not present in current row, current column and current 3X3 subgrid. After checking for safety, we assign the number, and recursively check whether this assignment leads to a solution or not. If the assignment doesn't lead to a solution, then we try next number for current empty cell. And if none of number (1 to 9) lead to solution, we return false.

ALGORITHM :

1. We send the input to rule 1 and the value of counter that counts total number of filled nodes
2. While (counter value increses)
    Apply rule 1
3. Apply Rule 2
4. If (counter value increses)
    Goto first step
5. Else
    Apply backtracking

RESULT : After the completion of above algorithm our final ouput satisfy all the four constrains listed above. And the approximate time taken by the algorithm to give output is 5 – 10 seconds.