LAB 4 : SUDOKO SOLVER

Procedure:

1. Each cell in puzzle contains at least one value.
2. Each cell in the puzzle contains at most one value.
3. Each row in the puzzle should contain all the values.
4. Each column in the puzzle should contain all the values.
5. Each smaller block should contain all the values.
6. The initial setup (values for some of the cells).

1) Cell Constraints:

Each cell in the Sudoku grid must have exactly one number from 1 to 9 assigned to it.
Create a clause for each cell at position (i, j) that includes literals representing all possible values: value(i, j, 1) OR value(i, j, 2) OR ... OR value(i, j, 9).
If a cell already contains a fixed value v, add a unit clause to enforce that specific value: value(i, j, v).

2)Pairwise Exclusion Constraints:

Guarantee that each cell in the Sudoku grid contains only one unique value.
Create clauses for each cell (i, j) to ensure that no two different values v1 and v2 are both assigned to that cell: -value(i, j, v1) OR -value(i, j, v2), for all distinct values v1 and v2.

3)Row and Column Constraints:

Ensure that each value from 1 to 9 appears exactly once in each row and each column.
Create clauses for each row and each column to ensure that no two cells in the same row or column contain the same digit.

For each row i and each column j, create clauses that ensure no two cells (i, j) and (i', j) or (i, j) and (i, j') contain the same value.

Example for rows: -value(i, j, v) OR -value(i', j, v) for all $i \neq i'$ and all possible values v.
Example for columns: -value(i, j, v) OR -value(i, j', v) for all $j \neq j'$ and all possible values v.

4)Square Region (3x3) Constraints:

Guarantee that each value from 1 to 9 appears exactly once in each 3x3 sub-grid.
Divide the Sudoku grid into 9 non-overlapping 3x3 sub-grids.

For each sub-grid, create clauses to ensure that no two cells within the same sub-grid contain the same value.

Example: For each sub-grid at position (i, j), create clauses such as -value(i, j, v) OR -value(i', j', v) for all cells (i, j) and (i', j') within the same sub-grid and for all possible values v.

By combining these clauses, we can represent the constraints of Sudoku as a Boolean SAT problem. Solving this SAT problem will yield a satisfying assignment, which corresponds to a valid solution to the Sudoku puzzle.