

Report_AI_LAB_4

Approach -

Let $x(i, j, v)$ represents the clause in i th row, j th column and v value

- Each entry has at least one value: $C_1 = \bigwedge_{r,c \in [1..n]} (x_{r,c,1} \vee x_{r,c,2} \vee \dots \vee x_{r,c,n})$
- Each entry has at most one value: $C_2 = \bigwedge_{r,c,v,v' \in [1..n] \text{ with } v < v'} (\neg x_{r,c,v} \vee \neg x_{r,c,v'})$
- Each row has all the numbers: $C_3 = \bigwedge_{r,v \in [1..n]} (x_{r,1,v} \vee x_{r,2,v} \vee \dots \vee x_{r,n,v})$
- Each column has all the numbers: $C_4 = \bigwedge_{c,v \in [1..n]} (x_{1,c,v} \vee x_{2,c,v} \vee \dots \vee x_{n,c,v})$
- Each disjoint sub-grid has all the numbers: $C_5 = \bigwedge_{r',c' \in [1..d], v \in [1..n]} (\bigvee_{(r,c) \in B_d(r'-1, c'-1)} x_{r,c,v})$
where $B_d(r', c') = \{(r'd + i, c'd + j) \mid i, j \in [1..d]\}$
- The solution respects the given clues H : $C_6 = \bigwedge_{(r,c,v) \in H} (x_{r,c,v})$

In my implementation (x,y,v) is tuple which represents -

- 1) x th row (1 based)
- 2) y th row (1 based)
- 3) v value stored in (x,y) cell (1 based)
- 4) $(-x,-y,-z)$ represents $\neg(x,y,z)$

Then we have created a dictionary d to store key as clause and unique id as a value for that clause for pycosat to work on it

We also created a dictionary d_rev to restore which unique id corresponds to which clause

Output -

The output is written to `sol.txt` in the same folder where `sudo_sol.py` is present.

Each line represents a solution corresponding to the input text file.