Experiment No.: 4

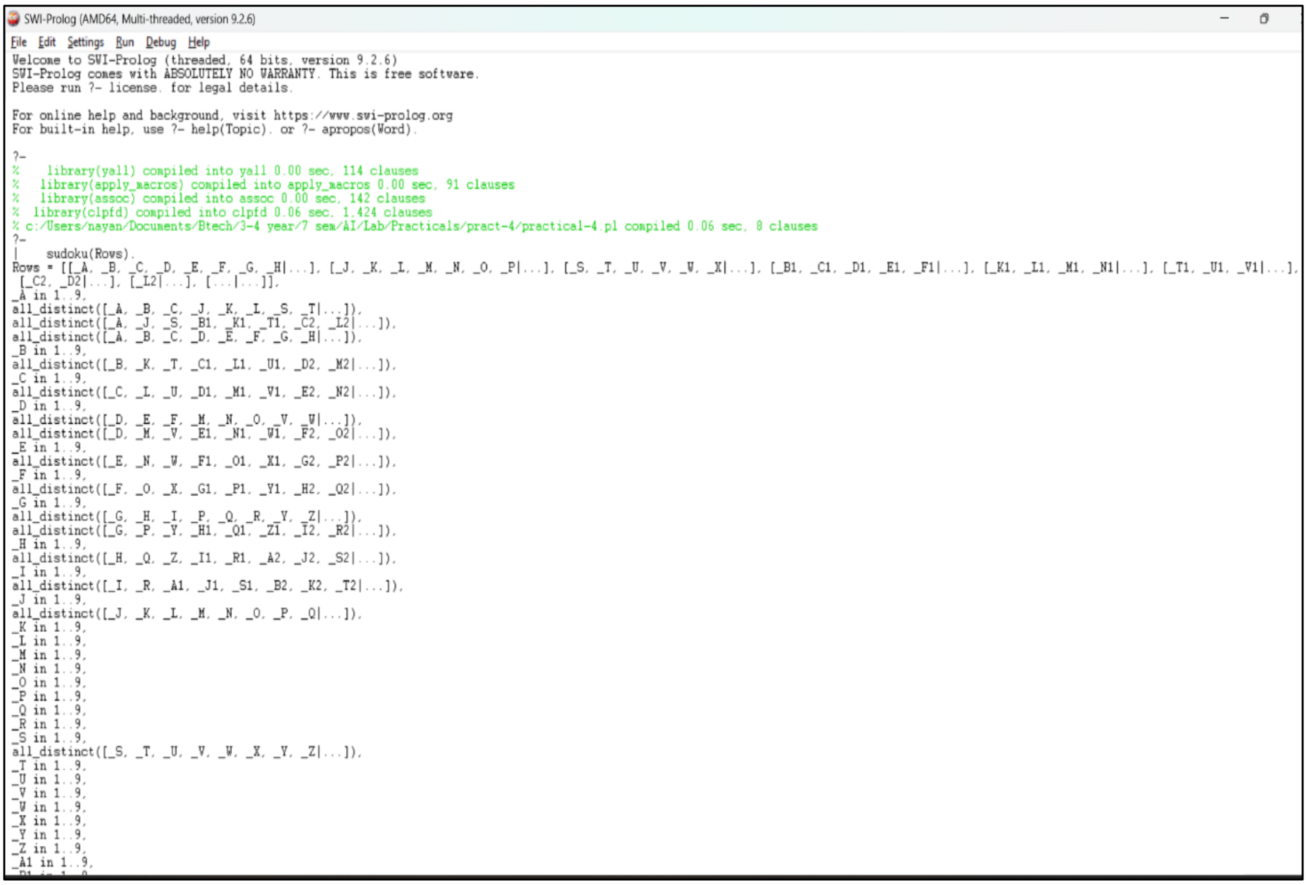
Solve Sudoku using Prolog

Code:

```
% to run the code in SWI-Prolog, do

% problem(1, Rows), sudoku(Rows), maplist(portray_clause, Rows).

:- use_module(library(clpfd)).

sudoku(Rows) :-

    length(Rows, 9), maplist(same_length(Rows), Rows),

    append(Rows, Vs), Vs ins 1..9,

    maplist(all_distinct, Rows),

    transpose(Rows, Columns),

    maplist(all_distinct, Columns),

    Rows = [As,Bs,Cs,Ds,Es,Fs,Gs,Hs,Is],

    blocks(As, Bs, Cs),

blocks(Ds, Es, Fs),

    blocks(Gs, Hs, Is).


blocks([], [], []).

blocks([N1,N2,N3|Ns1], [N4,N5,N6|Ns2], [N7,N8,N9|Ns3]) :-

    all_distinct([N1,N2,N3,N4,N5,N6,N7,N8,N9]),

    blocks(Ns1, Ns2, Ns3).


problem(0, [[_,_,_,_,_,_,_,_,_],

      [_,_,_,_,_,3,_,8,5],

      [_,_,1,_,2,_,_,_,_],

      [_,_,_,5,_,7,_,_,_],

      [_,_,4,_,_,_,1,_,_],

      [_,9,_,_,_,_,_,_,_],

      [5,_,_,_,_,_,_,7,3],

      [_,_,2,_,1,_,_,_,_],

      [_,_,_,_,4,_,_,_,9]]).
```

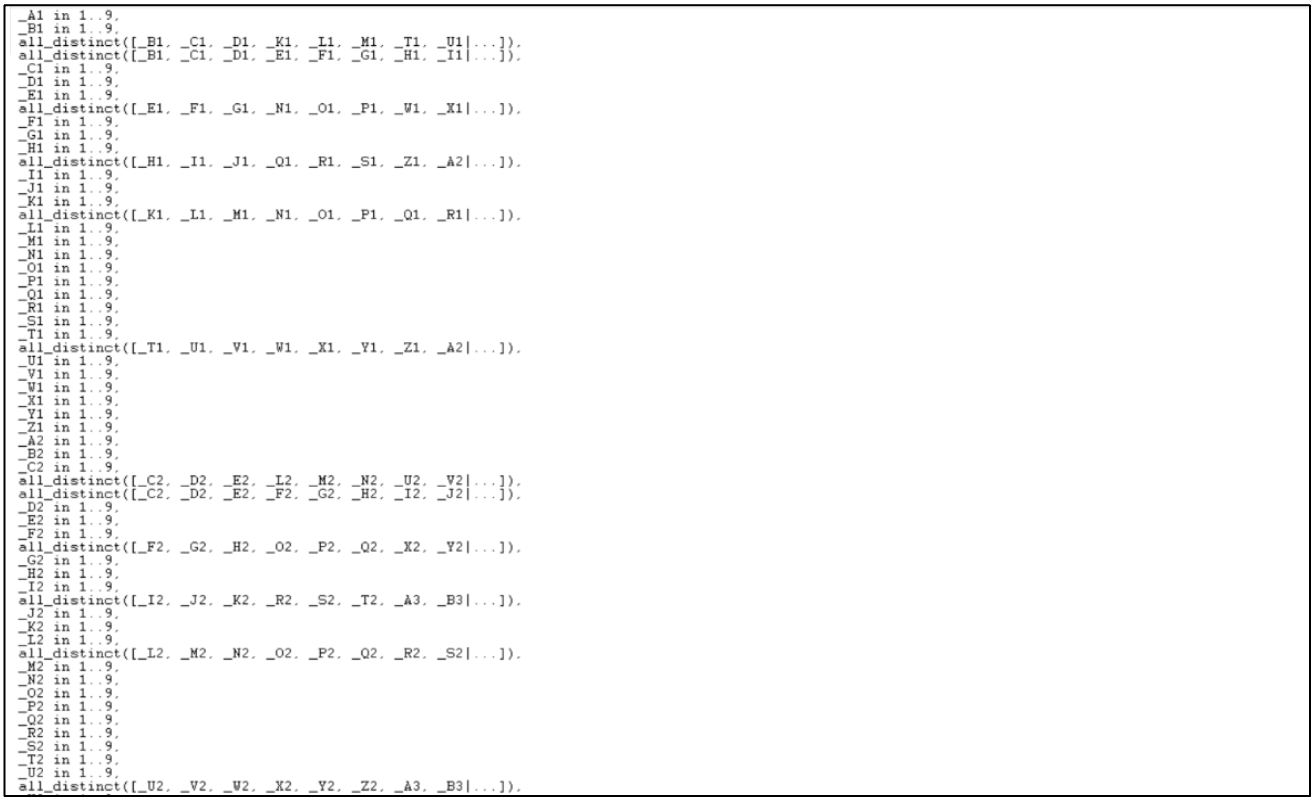
```
problem(1, P) :-
P = [[1,_,_,8,_,4,_,_,_],
          [_,2,_,_,_,_,4,5,6],
          [_,_,3,2,_,5,_,_,_],
          [_,_,_,4,_,_,8,_,5],
          [7,8,9,_,5,_,_,_,_],
          [_,_,_,_,_,6,2,_,3],
          [8,_,1,_,_,_,7,_,_],
          [_,_,_,1,2,3,_,8,_],
          [2,_,5,_,_,_,_,_,9]].

problem(2, P) :-
      P = [[_,_,2,_,3,_,1,_,_],
          [_,4,_,_,_,_,_,3,_],
          [1,_,5,_,_,_,_,8,2],
          [_,_,_,2,_,_,6,5,_],
          [9,_,_,_,8,7,_,_,3],
          [_,_,_,_,4,_,_,_,_],
          [8,_,_,_,7,_,_,_,4],
          [_,9,3,1,_,_,_,6,_],
          [_,_,7,_,6,_,5,_,_]].

problem(3, P) :-
      P = [[1,_,_,_,_,_,_,_,_],
          [_,_,2,7,4,_,_,_,_],
          [_,_,_,5,_,_,_,_,4],
          [_,3,_,_,_,_,_,_,_],
          [7,5,_,_,_,_,_,_,_],
          [_,_,_,_,_,9,6,_,_],
          [_,4,_,_,_,6,_,_,_],
          [_,_,_,_,_,_,_,7,1],
          [_,_,_,_,_,1,_,3,_]].
```

Output:

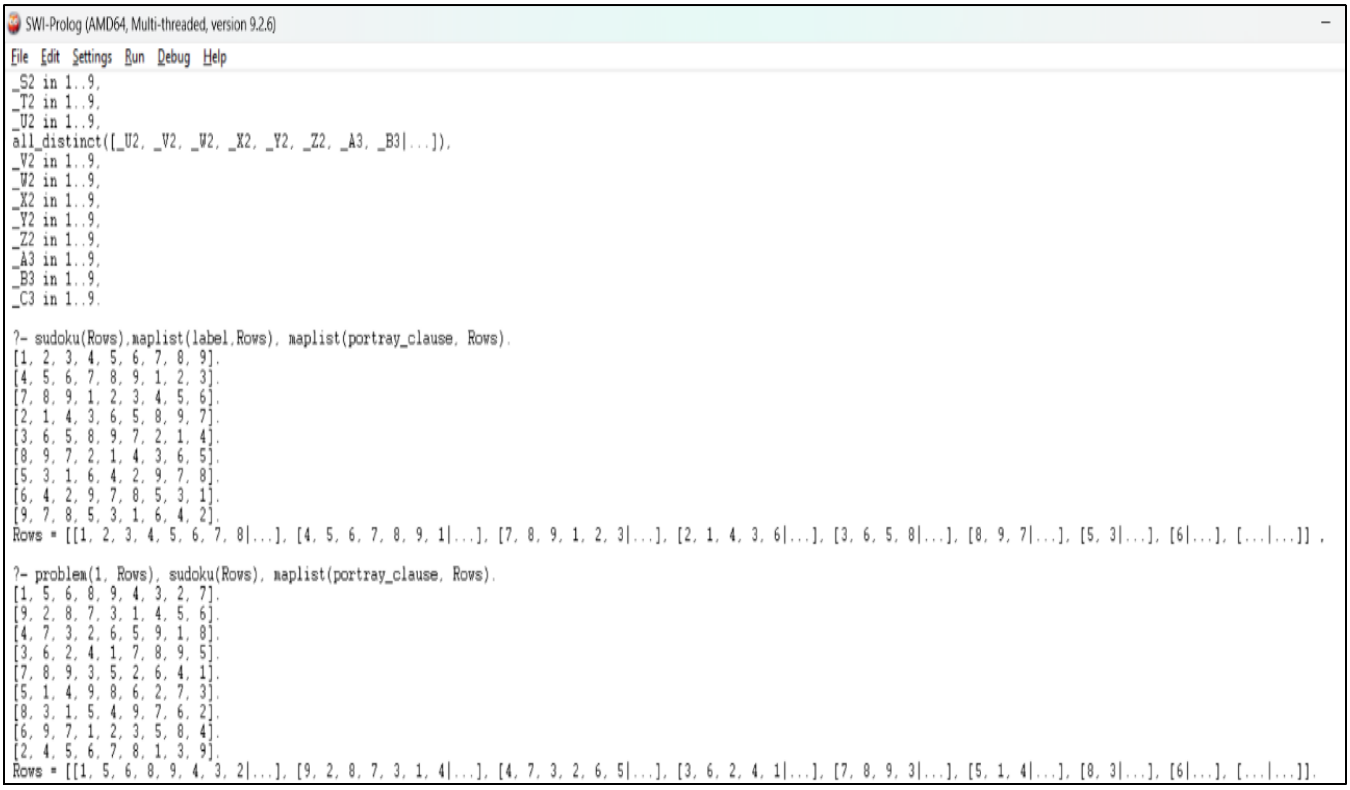


SWI-Prolog (AMD64, Multi-threaded, version 9.2.6)
File Edit Settings Run Debug Help

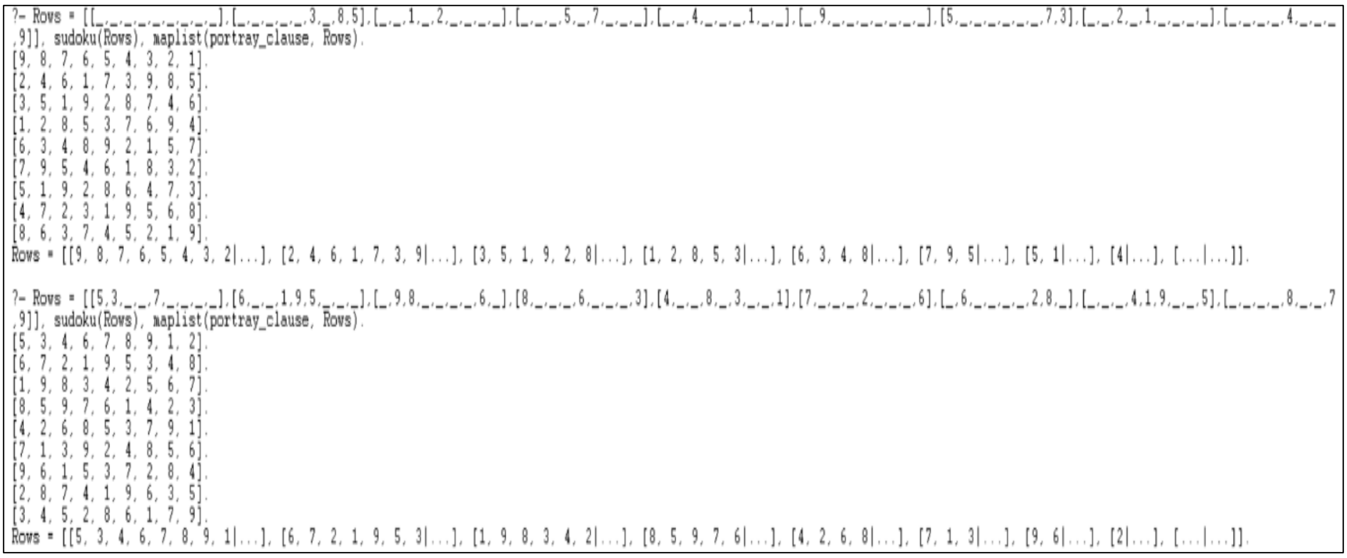
```
Welcome to SWI-Prolog (threaded, 64 bits, version 9.2.6)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
%    library(yall) compiled into yall 0.00 sec, 114 clauses
%   library(apply_macros) compiled into apply_macros 0.00 sec, 91 clauses
%   library(assoc) compiled into assoc 0.00 sec, 142 clauses
%  library(clpfd) compiled into clpfd 0.06 sec, 1,424 clauses
% c:/Users/nayan/Documents/Btech/3-4 year/7 sem/AI/Lab/Practicals/pract-4/practical-4.pl compiled 0.06 sec, 8 clauses
?-
|    sudoku(Rows).
Rows = [[_A, _B, _C, _D, _E, _F, _G, _H|...], [_J, _K, _L, _M, _N, _O, _P|...], [_S, _T, _U, _V, _W, _X|...], [_B1, _C1, _D1, _E1, _F1|...], [_K1, _L1, _M1, _N1|...], [_T1, _U1, _V1|...],
 [_C2, _D2|...], [_L2|...], [...|...]],
_A in 1..9,
all_distinct([_A, _B, _C, _J, _K, _L, _S, _T|...]),
all_distinct([_A, _J, _S, _B1, _K1, _T1, _C2, _L2|...]),
all_distinct([_A, _B, _C, _D, _E, _F, _G, _H|...]),
_B in 1..9,
all_distinct([_B, _K, _T, _C1, _L1, _U1, _D2, _M2|...]),
_C in 1..9,
all_distinct([_C, _L, _U, _D1, _M1, _V1, _E2, _N2|...]),
_D in 1..9,
all_distinct([_D, _E, _F, _M, _N, _O, _V, _W|...]),
all_distinct([_D, _M, _V, _E1, _N1, _W1, _F2, _O2|...]),
_E in 1..9,
all_distinct([_E, _N, _W, _F1, _O1, _X1, _G2, _P2|...]),
_F in 1..9,
all_distinct([_F, _O, _X, _G1, _P1, _Y1, _H2, _Q2|...]),
_G in 1..9,
all_distinct([_G, _H, _I, _P, _Q, _R, _Y, _Z|...]),
all_distinct([_G, _P, _Y, _H1, _Q1, _Z1, _I2, _R2|...]),
_H in 1..9,
all_distinct([_H, _Q, _Z, _I1, _R1, _A2, _J2, _S2|...]),
_I in 1..9,
all_distinct([_I, _R, _A1, _J1, _S1, _B2, _K2, _T2|...]),
_J in 1..9,
all_distinct([_J, _K, _L, _M, _N, _O, _P, _Q|...]),
_K in 1..9,
_L in 1..9,
_M in 1..9,
_N in 1..9,
_O in 1..9,
_P in 1..9,
_Q in 1..9,
_R in 1..9,
_S in 1..9,
all_distinct([_S, _T, _U, _V, _W, _X, _Y, _Z|...]),
_T in 1..9,
_U in 1..9,
_V in 1..9,
_W in 1..9,
_X in 1..9,
_Y in 1..9,
_Z in 1..9,
_A1 in 1..9,
```



```
 _A1 in 1..9,
 _B1 in 1..9,
 all_distinct([_B1, _C1, _D1, _K1, _L1, _M1, _T1, _U1|...]),
 all_distinct([_B1, _C1, _D1, _E1, _F1, _G1, _H1, _I1|...]),
 _C1 in 1..9,
 _D1 in 1..9,
 _E1 in 1..9,
 all_distinct([_E1, _F1, _G1, _N1, _O1, _P1, _W1, _X1|...]),
 _F1 in 1..9,
 _G1 in 1..9,
 _H1 in 1..9,
 all_distinct([_H1, _I1, _J1, _Q1, _R1, _S1, _Z1, _A2|...]),
 _I1 in 1..9,
 _J1 in 1..9,
 _K1 in 1..9,
 all_distinct([_K1, _L1, _M1, _N1, _O1, _P1, _Q1, _R1|...]),
 _L1 in 1..9,
 _M1 in 1..9,
 _N1 in 1..9,
 _O1 in 1..9,
 _P1 in 1..9,
 _Q1 in 1..9,
 _R1 in 1..9,
 _S1 in 1..9,
 _T1 in 1..9,
 all_distinct([_T1, _U1, _V1, _W1, _X1, _Y1, _Z1, _A2|...]),
 _U1 in 1..9,
 _V1 in 1..9,
 _W1 in 1..9,
 _X1 in 1..9,
 _Y1 in 1..9,
 _Z1 in 1..9,
 _A2 in 1..9,
 _B2 in 1..9,
 _C2 in 1..9,
 all_distinct([_C2, _D2, _E2, _L2, _M2, _N2, _U2, _V2|...]),
 all_distinct([_C2, _D2, _E2, _F2, _G2, _H2, _I2, _J2|...]),
 _D2 in 1..9,
 _E2 in 1..9,
 _F2 in 1..9,
 all_distinct([_F2, _G2, _H2, _O2, _P2, _Q2, _X2, _Y2|...]),
 _G2 in 1..9,
 _H2 in 1..9,
 _I2 in 1..9,
 all_distinct([_I2, _J2, _K2, _R2, _S2, _T2, _A3, _B3|...]),
 _J2 in 1..9,
 _K2 in 1..9,
 _L2 in 1..9,
 all_distinct([_L2, _M2, _N2, _O2, _P2, _Q2, _R2, _S2|...]),
 _M2 in 1..9,
 _N2 in 1..9,
 _O2 in 1..9,
 _P2 in 1..9,
 _Q2 in 1..9,
 _R2 in 1..9,
 _S2 in 1..9,
 _T2 in 1..9,
 _U2 in 1..9,
 all_distinct([_U2, _V2, _W2, _X2, _Y2, _Z2, _A3, _B3|...]),
```


SWI-Prolog (AMD64, Multi-threaded, version 9.2.6)
File Edit Settings Run Debug Help

```
_S2 in 1..9,
_T2 in 1..9,
_U2 in 1..9,
all_distinct([_U2, _V2, _W2, _X2, _Y2, _Z2, _A3, _B3|...]),
_V2 in 1..9,
_W2 in 1..9,
_X2 in 1..9,
_Y2 in 1..9,
_Z2 in 1..9,
_A3 in 1..9,
_B3 in 1..9,
_C3 in 1..9.

?- sudoku(Rows),maplist(label,Rows), maplist(portray_clause, Rows).
[1, 2, 3, 4, 5, 6, 7, 8, 9].
[4, 5, 6, 7, 8, 9, 1, 2, 3].
[7, 8, 9, 1, 2, 3, 4, 5, 6].
[2, 1, 4, 3, 6, 5, 8, 9, 7].
[3, 6, 5, 8, 9, 7, 2, 1, 4].
[8, 9, 7, 2, 1, 4, 3, 6, 5].
[5, 3, 1, 6, 4, 2, 9, 7, 8].
[6, 4, 2, 9, 7, 8, 5, 3, 1].
[9, 7, 8, 5, 3, 1, 6, 4, 2].
Rows = [[1, 2, 3, 4, 5, 6, 7, 8|...], [4, 5, 6, 7, 8, 9, 1|...], [7, 8, 9, 1, 2, 3|...], [2, 1, 4, 3, 6|...], [3, 6, 5, 8|...], [8, 9, 7|...], [5, 3|...], [6|...], [...|...]] .

?- problem(1, Rows), sudoku(Rows), maplist(portray_clause, Rows).
[1, 5, 6, 8, 9, 4, 3, 2, 7].
[9, 2, 8, 7, 3, 1, 4, 5, 6].
[4, 7, 3, 2, 6, 5, 9, 1, 8].
[3, 6, 2, 4, 1, 7, 8, 9, 5].
[7, 8, 9, 3, 5, 2, 6, 4, 1].
[5, 1, 4, 9, 8, 6, 2, 7, 3].
[8, 3, 1, 5, 4, 9, 7, 6, 2].
[6, 9, 7, 1, 2, 3, 5, 8, 4].
[2, 4, 5, 6, 7, 8, 1, 3, 9].
Rows = [[1, 5, 6, 8, 9, 4, 3, 2|...], [9, 2, 8, 7, 3, 1, 4|...], [4, 7, 3, 2, 6, 5|...], [3, 6, 2, 4, 1|...], [7, 8, 9, 3|...], [5, 1, 4|...], [8, 3|...], [6|...], [...|...]].
```



```
?- Rows = [[_,_,_,_,_,_,_,_,_],[_,_,_,_,_,3,_,8,5],[_,_,1,_,2,_,_,_,_],[_,_,_,5,_,7,_,_,_],[_,_,4,_,_,_,1,_,_],[_,9,_,_,_,_,_,_,_],[5,_,_,_,_,_,_,7,3],[_,_,2,_,1,_,_,_,_],[_,_,_,_,4,_,_,_
,9]], sudoku(Rows), maplist(portray_clause, Rows).
[9, 8, 7, 6, 5, 4, 3, 2, 1].
[2, 4, 6, 1, 7, 3, 9, 8, 5].
[3, 5, 1, 9, 2, 8, 7, 4, 6].
[1, 2, 8, 5, 3, 7, 6, 9, 4].
[6, 3, 4, 8, 9, 2, 1, 5, 7].
[7, 9, 5, 4, 6, 1, 8, 3, 2].
[5, 1, 9, 2, 8, 6, 4, 7, 3].
[4, 7, 2, 3, 1, 9, 5, 6, 8].
[8, 6, 3, 7, 4, 5, 2, 1, 9].
Rows = [[9, 8, 7, 6, 5, 4, 3, 2|...], [2, 4, 6, 1, 7, 3, 9|...], [3, 5, 1, 9, 2, 8|...], [1, 2, 8, 5, 3|...], [6, 3, 4, 8|...], [7, 9, 5|...], [5, 1|...], [4|...], [...|...]].

?- Rows = [[5,3,_,_,7,_,_,_,_],[6,_,_,1,9,5,_,_,_],[_,9,8,_,_,_,_,6,_],[8,_,_,_,6,_,_,_,3],[4,_,_,8,_,3,_,_,1],[7,_,_,_,2,_,_,_,6],[_,6,_,_,_,_,2,8,_],[_,_,_,4,1,9,_,_,5],[_,_,_,_,8,_,_,7
,9]], sudoku(Rows), maplist(portray_clause, Rows).
[5, 3, 4, 6, 7, 8, 9, 1, 2].
[6, 7, 2, 1, 9, 5, 3, 4, 8].
[1, 9, 8, 3, 4, 2, 5, 6, 7].
[8, 5, 9, 7, 6, 1, 4, 2, 3].
[4, 2, 6, 8, 5, 3, 7, 9, 1].
[7, 1, 3, 9, 2, 4, 8, 5, 6].
[9, 6, 1, 5, 3, 7, 2, 8, 4].
[2, 8, 7, 4, 1, 9, 6, 3, 5].
[3, 4, 5, 2, 8, 6, 1, 7, 9].
Rows = [[5, 3, 4, 6, 7, 8, 9, 1|...], [6, 7, 2, 1, 9, 5, 3|...], [1, 9, 8, 3, 4, 2|...], [8, 5, 9, 7, 6|...], [4, 2, 6, 8|...], [7, 1, 3|...], [9, 6|...], [2|...], [...|...]].
```