

```

1
2 % A teacher wishes to seat 16 quarrelsome children
3 % in a 4x4 array of chairs
4 % Some of the children don't get along
5 % The problem is to seat the children so no student
6 % is seated adjacent (4 way adjacent) to a child
7 % they quarrel with
8 %
9 % The children are numbered 1 thru 16
10 % 1..8 are girls, 9..16 are boys
11 %
12 %
13 :- use_module(library(clpfd)).
14
15
16 compatible(Neighbor , Student) :-
17     Student #= 3   #==> Neighbor #< 9,% #3 doesn't want to sit next to icky boys
18     Student #= 5   #==> Neighbor #< 9,% #5 agrees with #3
19     Student #= 2   #==> Neighbor #\= 3,% #2 and #3 don't get along
20     Student #= 10  #==> Neighbor #> 8.% #10 doesn't like gross girls
21
22 constrain_up(1 , _ , _ , _).
23 constrain_up(R , C , Student , Board) :-
24     R > 1,
25     NR is R - 1,
26     member(seat(NR , C , Neighbor) , Board),
27     compatible(Neighbor , Student).
28
29 constrain_down(4 , _ , _ , _).
30 constrain_down(R , C , Student , Board) :-
31     R < 4,
32     NR is R + 1,
33     member(seat(NR , C , Neighbor) , Board),
34     compatible(Neighbor , Student).
35
36 constrain_left(_ , 1 , _ , _).
37 constrain_left(R , C , Student , Board) :-
38     C > 1,
39     NC is C - 1,
40     member(seat(R , NC , Neighbor) , Board),
41     compatible(Neighbor , Student).
42
43 constrain_right(_ , 4 , _ , _).
44 constrain_right(R , C , Student , Board) :-
45     C < 4,
46     NC is C + 1,
47     member(seat(R , NC , Neighbor) , Board),
48     compatible(Neighbor , Student).
49
50 constrain_pupil(Board , seat(R , C , Student)) :-
51     constrain_up(R , C , Student , Board),
52     constrain_down(R , C , Student , Board),
53     constrain_left(R , C , Student , Board),
54     constrain_right(R , C , Student , Board).

```

```

55
56 make_seat(R , C , seat(R , C , Student)) :-
57     Student in 1..16.
58
59 % map between seat(r,c,s) and raw variable
60 seat_student(seat(_R, _C, S) , S).
61
62 % map between the [seat(1,1,S1)...] representaion and the [S1]
63 % representation
64 board_students(In , _SoFar , Raw) :-
65     maplist(seat_student , In , Raw).
66
67 /*
68     How to map thru by hand 8cD
69
70 board_students([], _ , []).
71 board_students([seat(_, _ , S)|T] , _ , [S|Vs]) :-
72     board_students(T , _ , Vs).
73 */
74
75 make_board(Board) :-
76     findall(S ,
77         ( member(R , [1,2,3,4]) ,
78           member(C , [1,2,3,4]) ,
79           make_seat(R , C , S)) ,
80         Board),
81     maplist(seat_student , Board , Raw),
82     all_distinct(Raw).
83
84 write_board(Board) :-
85     member(R , [1,2,3,4]),
86     nl,
87     member(C , [1,2,3,4]),
88     member(seat(R, C, S), Board),
89     write(S), write(' '),
90     fail.
91 write_board(_) :- nl.
92
93 assign_all_pupils :-
94     make_board(Board),
95     maplist(constrain_pupil(Board) , Board),
96     maplist(seat_student , Board , Raw),
97     labeling([], Raw),
98     write_board(Board).

```