

AI PRACTICAL NO. 8

Girish Nhavkar 9560 TE COMPS A

1. Prolog for tower of Hanoi

```
% hanoi(+N, +A, +B, +C, -Moves)
```

```
% N: number of disks
```

```
% A: the name of the source peg
```

```
% B: the name of the auxiliary peg
```

```
% C: the name of the destination peg
```

```
% Moves: list of moves to solve the Tower of Hanoi problem
```

```
hanoi(0, _, _, _, []).
```

```
hanoi(N, A, B, C, Moves) :-
```

```
    N > 0,
```

```
    N1 is N - 1,
```

```
    hanoi(N1, A, C, B, Moves1),
```

```
    append(Moves1, [(A,C)], Moves2),
```

```
    hanoi(N1, B, A, C, Moves3),
```

```
    append(Moves2, Moves3, Moves).
```

```
% Example usage:
```

```
% ?- hanoi(3, left, middle, right, Moves).
```

```
% Moves = [(left, right), (left, middle), (right, middle), (left, right), (middle, left), (middle, right), (left, right)].
```

```
% ?- hanoi_moves(4, Count).
```

```
% Count = 15
```

```
% ?- between(1, 5, N), hanoi(N, left, middle, right, Moves), writeln(Moves).
```

```
% [(left,right)]
```

```
%Moves = [(left,right)],
```

```
%N = 1
```

% If you only want to count the number of moves without storing them:

```
% hanoi_moves(+N, -Count)
```

```
hanoi_moves(N, Count) :-
```

```
    hanoi(N, _, _, _, Moves),
```

```
    length(Moves, Count).
```

OUTPUT:

```
1 % hanoi(+N, +A, +B, +C, -Moves)
2 % N: number of disks
3 % A: the name of the source peg
4 % B: the name of the auxiliary peg
5 % C: the name of the destination peg
6 % Moves: List of moves to solve the Tower of Hanoi problem
7 hanoi(0, _, _, _, []).
8 hanoi(N, A, B, C, Moves) :-
9     N > 0,
10    N1 is N - 1,
11    hanoi(N1, A, C, B, Moves1),
12    append(Moves1, [(A,C)], Moves2),
13    hanoi(N1, B, A, C, Moves3),
14    append(Moves2, Moves3, Moves).
15
16 % Example usage:
17 ?- hanoi(3, left, middle, right, Moves).
18 % Moves = [(left, right), (left, middle), (right, middle), (left, right), (middle, left), (middle, right), (left, right)]
19
20 ?- hanoi_moves(4, Count).
21 % Count = 15
22
23 ?- between(1, 5, N), hanoi(N, left, middle, right, Moves), writeln(Moves).
24 % [(left,right)]
25 % Moves = [(left,right)],
26 % N = 1
27
```

hanoi(3, left, middle, right, Moves).
Moves = [(left,right), (left,middle), (right,middle), (left,right), (middle,left), (middle,right), (left,right)]
false

hanoi_moves(4, Count).
Count = 15
Next 10 100 1,000 Stop

between(1, 5, N), hanoi(N, left, middle, right, Moves), writeln(Moves).
[(left,right)]
Moves = [(left,right)],
N = 1
Next 10 100 1,000 Stop

?- between(1, 5, N), hanoi(N, left, middle, right, Moves), writeln(Moves).

2. Prolog for N- Queen

```
% render solutions nicely.
```

```
:- use_rendering(chess).
```

```

%%    n_queens(?N, ?Cols) is nondet.
%
%    @param The k-th element of Cols is the column number of the
%    queen in row k.
%    @author Markus Triska

```

```

:- use_module(library(clpfd)).

```

```

n_queens(N, Qs) :-
    length(Qs, N),
    Qs ins 1..N,
    safe_queens(Qs).

```

```

safe_queens([]).
safe_queens([Q|Qs]) :-
    safe_queens(Qs, Q, 1),
    safe_queens(Qs).

```

```

safe_queens([], _, _).
safe_queens([Q|Qs], Q0, D0) :-
    Q0 #\= Q,
    abs(Q0 - Q) #\= D0,
    D1 #= D0 + 1,
    safe_queens(Qs, Q0, D1).

```

```

/** <examples>

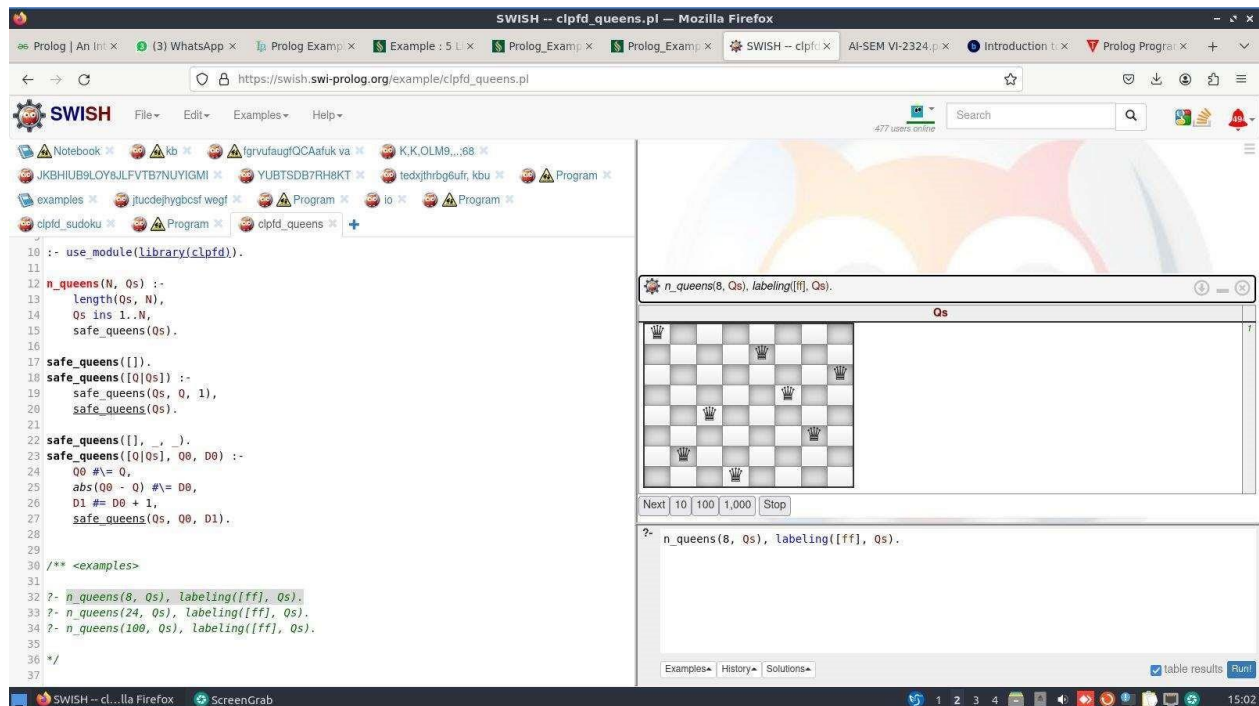
```

?- n_queens(8, Qs), labeling([ff], Qs).

?- n_queens(24, Qs), labeling([ff], Qs).

?- n_queens(100, Qs), labeling([ff], Qs).

*/



SWISH -- clpfd_queens.pl — Mozilla Firefox

Prolog | An In... | (3) WhatsApp | Prolog Exam... | Example: 5... | Prolog_Examp... | Prolog_Examp... | SWISH -- clpfd... | AI-SEM VI-2324... | Introduction... | Prolog Progra... | +

https://swish.swi-prolog.org/example/clpfd_queens.pl

SWISH File Edit Examples Help

477 users online

Search

clpfd_queens.pl

```
10 :- use_module(library(clpfd)).
11
12 n_queens(N, Qs) :-
13     length(Qs, N),
14     Qs ins 1..N,
15     safe_queens(Qs).
16
17 safe_queens([]).
18 safe_queens([Q|Qs]) :-
19     safe_queens(Qs, Q, 1),
20     safe_queens(Qs).
21
22 safe_queens([], _, _).
23 safe_queens([Q|Qs], Q0, D0) :-
24     Q0 #\= Q,
25     abs(Q0 - Q) #\= D0,
26     D1 #= D0 + 1,
27     safe_queens(Qs, Q0, D1).
28
29 /** <examples>
30
31 ?- n_queens(8, Qs), labeling([ff], Qs).
32 ?- n_queens(24, Qs), labeling([ff], Qs).
33 ?- n_queens(100, Qs), labeling([ff], Qs).
34
35 */
```

n_queens(8, Qs), labeling([ff], Qs).

Qs

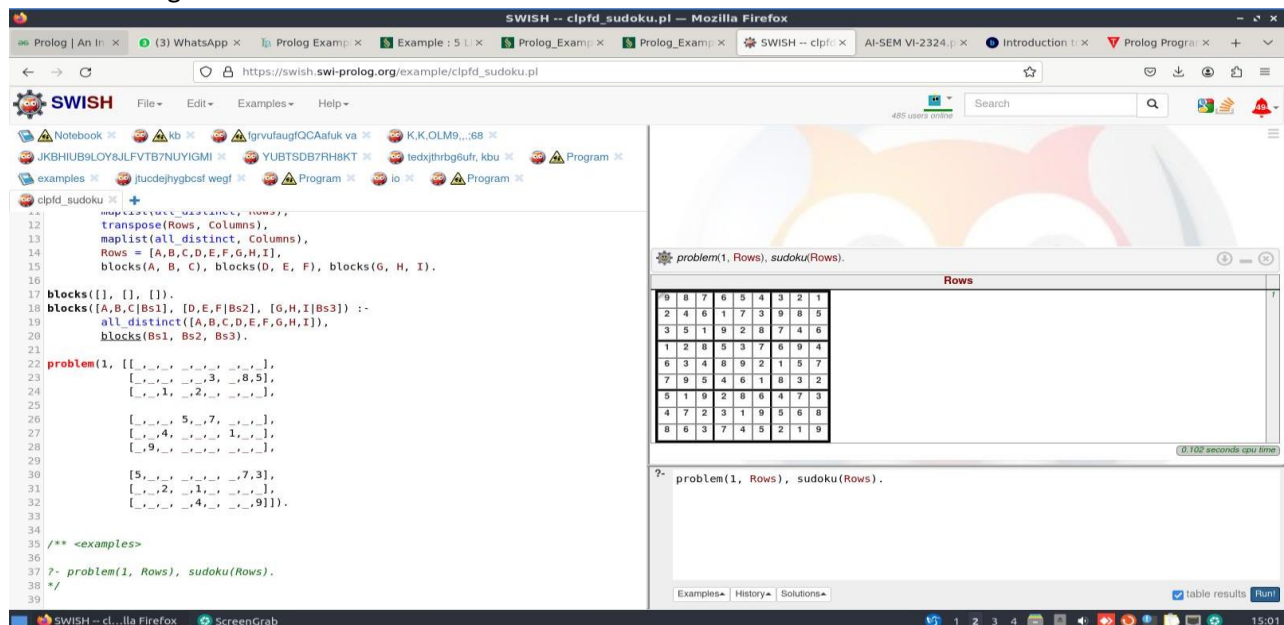
Next 10 100 1,000 Stop

?- n_queens(8, Qs), labeling([ff], Qs).

Examples History Solutions

table results Run

Some more eg:



SWISH -- clpfd_sudoku.pl — Mozilla Firefox

Prolog | An In... | (3) WhatsApp | Prolog Exam... | Example: 5... | Prolog_Examp... | Prolog_Examp... | SWISH -- clpfd... | AI-SEM VI-2324... | Introduction... | Prolog Progra... | +

https://swish.swi-prolog.org/example/clpfd_sudoku.pl

SWISH File Edit Examples Help

485 users online

Search

clpfd_sudoku.pl

```
10 maplist(all_distinct, Rows),
11 transpose(Rows, Columns),
12 maplist(all_distinct, Columns),
13 Rows = [A,B,C,D,E,F,G,H,I],
14 blocks(A, B, C), blocks(D, E, F), blocks(G, H, I).
15
16 blocks([], [], []).
17 blocks([A,B,C|Bs1], [D,E,F|Bs2], [G,H,I|Bs3]) :-
18     all_distinct([A,B,C,D,E,F,G,H,I]),
19     blocks(Bs1, Bs2, Bs3).
20
21 problem(1, [_,_,_,_,_,_,_,_,_],
22             [_,_,_,_,_,_,_,_,_],
23             [_,_,_,_,_,_,_,_,_],
24             [_,_,_,_,_,_,_,_,_],
25             [_,_,_,_,_,_,_,_,_],
26             [_,_,_,_,_,_,_,_,_],
27             [_,_,_,_,_,_,_,_,_],
28             [_,_,_,_,_,_,_,_,_],
29             [_,_,_,_,_,_,_,_,_],
30             [_,_,_,_,_,_,_,_,_],
31             [_,_,_,_,_,_,_,_,_],
32             [_,_,_,_,_,_,_,_,_]).
33
34 /** <examples>
35
36 ?- problem(1, Rows), sudoku(Rows).
37
38 */
```

problem(1, Rows), sudoku(Rows).

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

0.102 seconds cpu time

?- problem(1, Rows), sudoku(Rows).

Examples History Solutions

table results Run