AI ASSIGNMENT – 7

Write a PROLOG code to implement Maze.

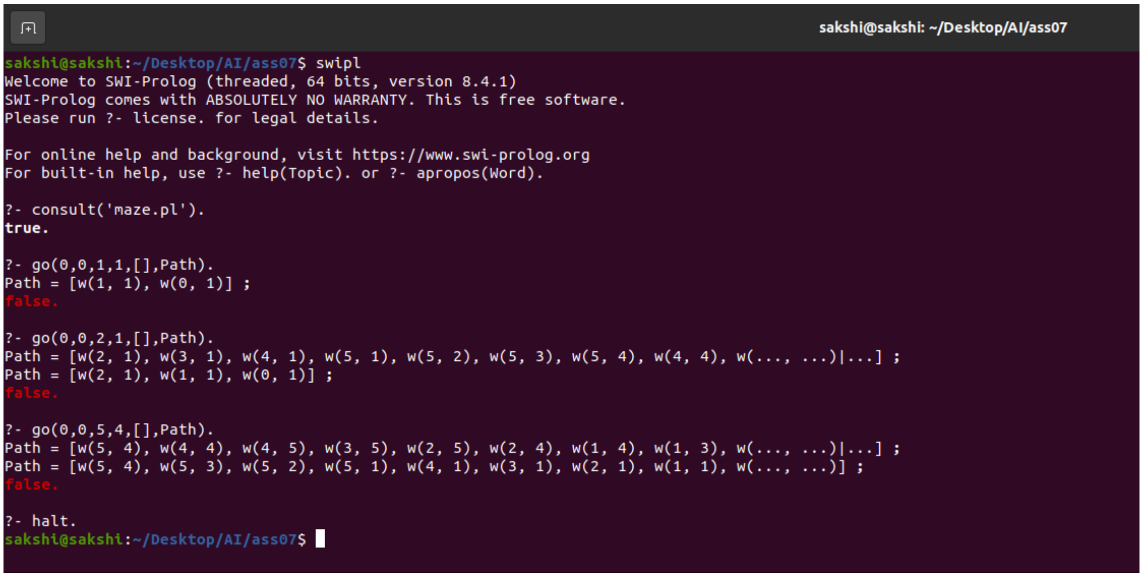
Source Code:

```
% Maze Problem in Prolog

w(0,0).
w(0,1). w(1,1). w(2,1). w(3,1). w(4,1). w(5,1).
        w(1,2).         w(3,2).         w(5,2).
        w(1,3).         w(3,3).         w(5,3).
w(0,4). w(1,4). w(2,4).         w(4,4). w(5,4).
                w(2,5). w(3,5). w(4,5).

d(X0,Y0,X,Y) :- next_w(X0,Y0,X,Y), w(X,Y).
next_w(X0,Y0,X0,Y) :- Y is Y0+1.
next_w(X0,Y0,X,Y0) :- X is X0+1.
next_w(X0,Y0,X0,Y) :- Y is Y0-1.
next_w(X0,Y0,X,Y0) :- X is X0-1.

go(X,Y,X,Y,Path,Path).
go(X0,Y0,X,Y,SoFar,Path) :-
    d(X0,Y0,X1,Y1),
    \+ memberchk( w(X1,Y1), SoFar ),
    go(X1,Y1,X,Y,[w(X1,Y1)|SoFar],Path).
```



```
sakshi@sakshi:~/Desktop/AI/ass07$ swipl
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
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).

?- consult('maze.pl').
true.

?- go(0,0,1,1,[],Path).
Path = [w(1, 1), w(0, 1)] ;
false.

?- go(0,0,2,1,[],Path).
Path = [w(2, 1), w(3, 1), w(4, 1), w(5, 1), w(5, 2), w(5, 3), w(5, 4), w(4, 4), w(..., ...)|...] ;
Path = [w(2, 1), w(1, 1), w(0, 1)] ;
false.

?- go(0,0,5,4,[],Path).
Path = [w(5, 4), w(4, 4), w(4, 5), w(3, 5), w(2, 5), w(2, 4), w(1, 4), w(1, 3), w(..., ...)|...] ;
Path = [w(5, 4), w(5, 3), w(5, 2), w(5, 1), w(4, 1), w(3, 1), w(2, 1), w(1, 1), w(..., ...)] ;
false.

?- halt.
sakshi@sakshi:~/Desktop/AI/ass07$ 
```