```
/* 1 */

e([], []).
e([[] | YS], ZS) :- e(YS, ZS),!.
e([[H | T] | YS], ZS) :- e([H | [T | YS]], ZS),!.
e([X | XS], [X | ZS]) :- e(XS, ZS).


/* 2 */

add(DS,V,RS) :-
  reverse(DS,RD),
  sumx(RD,V,[],RS).

sumx([D|DS],C,VS,XS) :-
  S is C+D, NC is S // 10, V is S - NC*10,
  sumx(DS,NC,[V|VS],XS).
sumx([],0,VS,VS) :- !.
sumx([],C,VS,XS) :-
  NC is C // 10, V is C - NC*10,
  sumx([],NC,[V|VS],XS).


/* 3 */

lookup(T,N,V,NT) :-
  var(T),!,fail.
lookup(T,N,V,NT) :-
  nonvar(N), nonvar(V), ins(T,N,V,NT), !.
lookup(T,N,V,NT) :-
  nonvar(N), tst(T,N,V), T=NT.

ins([], N, V, [(N,V)]).
ins([(N,_) | VS], N, V, [(N,V) | VS]) :- !.
ins([X|XS], N, V, [X|VS]) :- ins(XS, N, V, VS).

tst([], _, _) :- !, fail.
tst([(N,V) | _], N, V) :- !.
tst([_ | VS], N, V) :- tst(VS, N, V).


/* 4 */

search(P,LL) :-
  retractall(pos(_)),
  bagof(L,track(P,P,0,L),LL).

track(P,P,N,[P]) :- N >= 20, N =< 22, !.
track(P,P,N,_) :- (N>0,!, fail;N==0,fail).
track(A,P,N,[A|T]) :-
  N < 22,
  assertz(pos(A)),
  nextStep(A,B),
  (not(pos(B));B==P),
  NN is N+1,
  track(B,P,NN,T).
track(A,_,_,_) :-
  pos(A),
  retract(pos(A)),
  !, fail.


/* toto nebylo treba */

nextStep(p(X,Y),p(XX,YY)) :-
  move(I,J),
  XX is X+I,
  YY is Y+J,
  XX > 0, YY > 0.
  /* XX < 9, YY < 9. */

move(1,0).
move(0,1).
move(-1,0).
move(0,-1).


/* EOF */
```