Logic Programs as Specifications

```
fleck@tornado [101]% more list.laws
% Laws of lists ARE programs -- CLP(R) version

length([ ],0).
length([X|Xs],N+1) :- length(Xs, N).
% Xs,N,X . length(Xs, N) => length([X | Xs], N+1)

member(X,[X|Xs]).          % X,Xs member(X, [X | Xs])
member(X,[Y|Xs]) :- member(X,Xs).
% X,Y,Xs . member(X, Xs) => member(X, [Y | Xs])

select([X|Xs],1, X).          % X,Xs select([X | Xs] ,1, X)
select([X|Xs], N+1, Y) :- select(Xs, N, Y).
% X,Y,N,Xs . select(Xs, N, Y) => select([X | Xs],  N+1, Y)

insert( Y, Xs, 1, [Y|Xs]). % Y,Xs insert(Y, Xs, 1, [Y | Xs])
insert(Y, [X|Xs], N+1, [X|Ys]) :- N>0, insert(Y, Xs, N, Ys).
% X,Y,N,Xs,Ys . N>0 ∧ insert(Y, Xs, N, Ys) =>
%          insert(Y, [X|Xs], N+1, [X | Ys])
```

```
append([ ], Ys, Ys).           % Ys append([ ],Ys,Ys)
append([X|Xs], Ys, [X|Zs]) :- append(Xs, Ys, Zs).
% X,Xs,Ys,Zs . append(Xs,Ys,Zs) =>
                               % append([X|Xs], Ys, [X|Zs])


fleck@tornado [102]% clpr


CLP(R) Version 1.2
(c) Copyright International Business Machines
Corporation
1989 (1991, 1992) All Rights Reserved


1 ?- consult('list.laws').
Warning: Style check, singleton variables, rule 2 of
length/2
  +++  X
*** Yes


2 ?- member(b,[a,b,c]).

*** Retry?


3 ?- member(d,[a,b,c]).
*** No
```

```
4 ?- member(X,[a,b,c]).
  X = a
  *** Retry? ;
  X = b
  *** Retry? ;
  X = c
  *** Retry? ;

  *** No

  5 ?- member(b,[a,X,c]).
  X = b
*** Retry?

  6 ?- member(X,[a,b,c,d]), member(X,[c,d,e]).
  X = c
  *** Retry? ;
  X = d
  *** Retry? ;

  *** No

  7 ?- append([a,b], [c,d], As).
  As = [a, b, c, d]

  *** Yes

  8 ?- append(As, [c,d], [a,b,c,d]).
  As = [a, b]

  *** Retry?
```

9 ?- append(As, Bs, [a,b,c]).
Bs = [a, b, c]
As = [ ]

*** Retry? ;

Bs = [b, c]
As = [a]

*** Retry? ;

Bs = [c]
As = [a, b]

*** Retry? ;

Bs = [ ]
As = [a, b, c]

*** Retry? ;

*** No

10 ?- insert(c,[a, b, d], 3, As).
As = [a, b, c, d]

*** Retry?

11 ?- insert(c,[a,b,d],N,[a,b,c,d]).
N = 3

*** Retry?

```
 12 ?- select([a,b,c,d],3,X).
X = c

*** Retry? ;

*** No

13 ?- select([a,b,c,d],N,b).
N = 2

*** Retry? ;

*** No

14 ?- select([a,b,X,d],N,c).
X = c
N = 3

*** Retry?
fleck@tornado [103]% exit
   exit
   script done
```