

باسمه تعالی

پرو لوگ قسمت چهار

حل تمرین های قسمت قبل:

1- محمول permutation

```
permutation( [], []).  
permutation( L, [X|P]) :-  
    del( X, L, L1),  
    permutation( L1, P).
```

2- محمول reverse

```
reverse( [], []).  
reverse( [ First | Rest], Reversed) :-  
    conc( ReversedList, [First], Reversed),  
    reverse( Rest, ReversedList).
```

3- محمول maxList

The general strategy for finding the maximum of a list is to compute the maximum of the tail and then to compare the result with the head of the list. Whichever is bigger is the maximum of the whole list.

```
maxlist([X], X).  
maxlist([X|L], X):-maxlist(L, V), X>=V.  
maxlist([X|L], V):-maxlist(L, V), X<V.
```

maxList([-2,9,6],9)
↓
V=9,X=-2,L=[9,6]
↓
maxList([9,6],9) , -2 < 9
↓
X=9, L=[6]
↓
maxList([6],V) , 9 >= V , -2 < 9
↓
X=6
↓
9 >= 6 , -2 < 9

بررسی چند case-study

1- مساله سیاحت اسب در شطرنج Knight-tour

به چه طریقی یک اسب در صفحه مربعی $N \times N$ می تواند حرکت کند به شکلی که از تمامی خانه های صفحه تنها یک بار بگذرد .

% knights(N,Knights) :- Knights is a knight's tour on a $N \times N$ chessboard

knights(N,Knights) :- M is $N \times N - 1$, knights(N,M,[1/1],Knights).

% knights(N,M,Visited,Knights) :- the list of squares Visited must be

% extended by M further squares to give the solution Knights of the

% $N \times N$ chessboard knight's tour problem.

knights(_,0,Knights,Knights).

knights(N,M,Visited,Knights) :-

Visited = [X/Y|_],

jump(N,X/Y,U/V),

\+ memberchk(U/V,Visited),

M1 is M-1,

knights(N,M1,[U/V|Visited],Knights).

% jumps on an $N \times N$ chessboard from square A/B to C/D

jump(N,A/B,C/D) :-

jump_dist(X,Y),

C is A+X, C > 0, C =< N,

D is B+Y, D > 0, D =< N.

% jump distances

jump_dist(1,2).

jump_dist(2,1).

jump_dist(2,-1).

jump_dist(1,-2).

jump_dist(-1,-2).

jump_dist(-2,-1).

jump_dist(-2,1).

jump_dist(-1,2).

در کد فوق به این نکته دقت کنید که + \ نشاندهنده ی مفهوم not است، که در مورد آن در فصل مربوط به cut بیشتر توضیح خواهیم داد.

2- تعیین اینکه آیا عددی اول است یا خیر

% is_prime(P) :- P is a prime number

% (integer) (+)

is_prime(2).

is_prime(3).

is_prime(P) :- P > 3, P mod 2 =\= 0, \+ has_factor(P,3).

% has_factor(N,L) :- N has an odd factor F >= L.

% (integer, integer) (+,+)

has_factor(N,L) :- N mod L =:= 0.

has_factor(N,L) :- L * L < N, L2 is L + 2, has_factor(N,L2).

3- بررسی حدس گلدباخ

کریستیان گلدباخ یکی از ریاضیدانان مشهور روسی قرن هجدهم می باشد. علت اصلی شهرت این ریاضی دان حدسی است که وی در حوزه نظریه اعداد زده است. گلدباخ در نامه ای که در سال ۱۷۴۲ برای اولر نوشت مساله ای را مطرح کرد که هنوز توسط هیچ کس اثبات یا رد نشده است. حدس گلدباخ : هر عدد زوج مثبت بزرگتر مساوی چهار را می توان به صورت جمع دو عدد اول بیان کرد. برای مثال

$$۲۰ = ۱۷ + ۳$$

$$۲۲ = ۱۹ + ۳$$

$$۲۴ = ۱۷ + ۷$$

% goldbach(N,L) :- L is the list of the two prime numbers that

% sum up to the given N (which must be even).

% (integer,integer) (+,-)

goldbach(4,[2,2]) :- !.

goldbach(N,L) :- N mod 2 =:= 0, N > 4, goldbach(N,L,3).

goldbach(N,[P,Q],P) :- Q is N - P, is_prime(Q), !.

goldbach(N,L,P) :- P < N, next_prime(P,P1), goldbach(N,L,P1).

next_prime(P,P1) :- P1 is P + 2, is_prime(P1), !.

next_prime(P,P1) :- P2 is P + 2, next_prime(P2,P1).