% k-th element of a list

% kelem(R,lista,N).

kelem(X,[X|\_],1).

kelem(X,[\_|T],N):- M is N-1, kelem(X,T,M).

reverse([],[]).

reverse([H|T],A):- reverse(T,B),union(B,[H],A).

egal([],[]).

egal([H|T],[H|L]):- egal(T,L).

palindrome(L):- rev(L,M),egal(L,M),write(M).

accrev([H|T],A,R):- accrev(T,[H|A],R).

accrev([],A,A).

rev(L,R):- accrev(L,[],R).

%duplin

%duplin(L,N,R)

dupl(\_,0,[]).

dupl(X,N,[X|T]):- M is N-1, dupl(X,M,T), N>0.

duplin([],\_,[]).

duplin([H|T],N,R):- dupl(H,N,L),duplin(T,N,M), union(L,M,R).

%split(L,N,R1,R2).

split1(\_,0,[]).

split1([H|T],N,[H|L]):- N>0, M is N-1, split1(T,M,L).

split2([],\_,[],\_).

split2([H|T],N,[H|L],A):- A>=N, split2(T,N,L,A).

split2([\_|T],N,L,A):- A<N,B is A+1, split2(T,N,L,B).

split(L,N,R1,R2):- split1(L,N,R1),split2(L,N,R2,0).

% remove kth element

%remove(X,[a,b,c,d],2,R).

remove(H,[H|T],1,T).

remove(X,[H|T],N,[H|L]):- N>1, M is N-1, remove(X,T,M,L).

%insert kth element

%insert(alfa,[a,b,c,d],2,L).

insert(A,L,1,[A|L]).

insert(A,[H|T],N,[H|L]):- N>1, M is N-1, insert(A,T,M,L).

%range

%range(4,9,L).

range(X,X,[X]).

range(X,Y,[X|L]):- X<Y, M is X+1, range(M,Y,L).

%is prime?

cmmdc(A,0,A).

cmmdc(A,B,R):- B =\=0, X is A mod B, cmmdc(B,X,R).

is\_prim(\_,[]).

is\_prim(X,[H|T]):- is\_prim(X,T), cmmdc(X,H,1).

prime(2).

prime(X):- Y is X-1, range(2,Y,L), is\_prim(X,L).

%prime factors

prfct(1,\_,[]).

prfct(X,D,[D|T]):-D=<X, A is X mod D, A=:=0, B is X/D, prfct(B,D,T).

prfct(X,D,L):- D=<X, A is X mod D, A=\=0, B is D+1, prfct(X,B,L).

prime\_factors(X,[]):-X<2.

prime\_factors(X,L):- prfct(X,2,L).

%short prime\_factors

short\_prime\_factors(X,L):- prime\_factors(X,A), encode\_direct(A,L).

encode\_direct([],[]).

encode\_direct([X|Xs],[Z|Zs]) :- count(X,Xs,Ys,1,Z), encode\_direct(Ys,Zs).

count(X,[],[],1,X).

count(X,[],[],N,[X,N]) :- N > 1.

count(X,[Y|Ys],[Y|Ys],1,X) :- X \= Y.

count(X,[Y|Ys],[Y|Ys],N,[X,N]) :- N > 1, X \= Y.

count(X,[X|Xs],Ys,K,T) :- K1 is K + 1, count(X,Xs,Ys,K1,T).

% listaNelem(lista,N,lN).

listaNelem(\_,0,[]).

listaNelem(L,N,[H|T]):- N>0, member(H,L), M is N-1, listaNelem(L,M,T).

listeNelem(L,N,LL) :- bagof(M, listaNelem(L,N,M), LL).

connected(1,2).

connected(3,4).

connected(5,6).

connected(7,8).

connected(9,10).

connected(12,13).

connected(13,14).

connected(15,16).

connected(17,18).

connected(19,20).

connected(4,1).

connected(6,3).

connected(4,7).

connected(6,11).

connected(14,9).

connected(11,15).

connected(16,12).

connected(14,17).

connected(16,19).

path(X,Y,[X,Y]):- connected(X,Y).

path(X,Y,[X|L]):- connected(X,Z),path(Z,Y,L).

pathc(X,Y):- path(X,Y,L), length(L,X), X>0.