

INSERTAR UN ELEMENTO

insert(X,[],[X]).

insert(Elem,[Car|Cdr],[Elem,Car|Cdr] :- Elem < Car.

insert(Elem,[Car|Cdr],[Car|Elem]) :- insert(Elem,Cdr,X).

IGUAL

igual([],[]).

igual(X,X).

NUMERO PAR

par(0).

par(s(s(X))):-par(X).

NUMERO IMPAR

impar(s(0)).

impar(s(s(X))):-impar(X).

SUMA

suma(0,X,X).

suma(s(X),Y,s(Z)) :- suma(X,Y,Z).

MULTIPLICACION

mult(0,_,0).

mult(s(X),Y,R) :- mult(X,Y,Z), suma(Z,Y,R).

ES MIEMBRO

miembro(X,[X|_]).

miembro(X,[Car|Cdr]) :- miembro(X,Cdr).

MAYOR O IGUAL

mayor(_,0).

mayor(s(X),s(Y)) :- mayor(X,Y).

MAYOR O ESTRICTO

mayor(s(X),0).

mayor(s(X),s(Y)) :- mayor(X,Y).

UNION DE UN CONJUNTO

union([],C,C).

union([Car|Cdr],C,Union) :- miembro(Car,C),!,union(Cdr,C,Union).

union([Car|Cdr],C,[Car|Union]) :- union(Cdr,C,Union).

INTERSECCION DE UN CONJUNTO

intersec([],C,[]).

intersec([Car|Cdr],C,[Car|Inter]) :- miembro(Car,C),!,intersec(Cdr,C,Inter).

intersec([Car|Cdr],C,Inter) :- intersec(Cdr,C,Inter).

conj([]).

conj([Car|Cdr]) :- miembro(Car,Cdr),!,fail.

conj([_|Cdr]) :- conj(Cdr).

INCLUSION

inclusion([],X).

inclusion([Car|Cdr],C) :- miembro(Car,C),inclusion(Cdr,C).

APLANAR

aplanar([],[]).

aplanar(Atomo,[Atomo]):-atomic(Atomo),Atomo \== [].

aplanar([Car|Cdr],R):-aplanar(Car,CarA),
aplanar(Cdr,CdrA),
concatenar(CarA,CdrA,R).

CARTESIANO

cartesiano([],C,[]).

cartesiano([Car|Cdr],C,Resultado) :- linea(Car,Conj,Linea),
cartesiano(Cdr,Conjunto,Resto),
concatenar(Linea,Resto,Resultado).

linea(Elem,[],[]).

linea(Elem,[Car|Cdr],[[Elem,Car]|Resto]) :- linea(Elem,Cdr,Resto).

concat([],X,X).

concat([Car|Cdr],Lista,[Car|Resultado]) :- concat(Cdr,Lista,Resultado).

INVERTIR

invertir([],[]).

invertir([Car|Cdr],X) :- invertir(Cdr,R),concat(R,[Car],X).

EXPONENTE

suma(0,X,X).

suma(s(X),Y,s(Z)) :- suma(X,Y,Z).

mult(0,_,0).

mult(s(X),Y,R) :- mult(X,Y,Z), suma(Z,Y,R).

exp(_,0,s(0)).

exp(N,s(0),N).

exp(N1,s(N2),res) :- exp(N1,N2,Par), mult(Par,N1,res).