**Lists**

1. Write a Prolog program to find the sum of all numbers in a given list.

domains

list = integer\*

predicates

sum(list,integer)

clauses

sum([H]|T], S):- sum(T,S1), S=H+S1.

sum([],0).

Goal.sum([1,2,3],S)

S=6

1. Write a Prolog program to check whether an element is a member of a given list or not.

domains

list = integer\*

predicates

mem(list,integer)

clauses

mem([H]|T], X):- mem(T,X).

mem([X|T],X).

Goal.mem([1,2,3],3)

Yes

1. Write a Prolog program to find the length of a list.

domains

list = integer\*

predicates

len(list,integer)

clauses

len([H]|T], S):- S1=S+1,len(T,S1).

len([],S):-write(S).

lenstart([H|T]):-len([H|T],0)

Goal.lenstart([1,2,3])

3

1. Write a Prolog program to check whether an element is placed next to another element in a given list.
2. Write a Prolog program to find the first element of a given list.

domains

list = integer\*

predicates

FElement(list)

clauses

FElement([H|T]):-write(H).

Goal. FElement([1,2,3])

S=1

1. Write a Prolog program to find the last element of a given list.

domains

list=symbol\*

predicates

last(list)

clauses

last([X]):-

write(\"\\nLast element is : \"),

write(X).

last([Y|Tail]):-

last(Tail).

Goal. LElement ([1,2,3])

S=3

1. Write a Prolog program to delete elements from a list.

domains

X=integer

list = integer\*

predicates

delete(integer,list,list)

clauses

delete(E,[E|T],T)

delete(E,[H|T],[H|T1]):-delete(E,T,T1)

Goal. delete (2,[1,2,3],L)

L=[1,3]

1. Write a Prolog program to insert elements into a list.

domains

X=integer

list = integer\*

predicates

ins(integer,list,list)

clauses

ins(X,[],[X])

ins(X,[Y|T],[Y|T1]):-ins (X,T,T1)

Goal. ins(2,[1,3,4],L)

L=[1,3,4,2]

1. Write a Prolog program to append two lists.

domains

list = integer\*

predicates

append(list,list,list)

clauses

append([H|T],L,[H|T1]):-append(T,L,T1).

append([],L,L).

Goal.append([1,2,3],[4,5],L)

L=[1,2,3,4,5]

1. Write a recursive program in Prolog to multiply the elements of integer list.

domains

list = integer\*

M=integer

predicates

mul(list, integer)

clauses

mul([H|T], M):-mul(T, M1), M=M1\*H.

mul([],L).

Goal.mul([1,2,3],M)

M = 6

1. Write a Prolog program to find the union of two given lists.
2. Write a Prolog program to find the intersection of two given lists.

1. Write a Prolog program to find reverse of a given list.

domains

list = integer\*

predicates

reverse(list,list,list)

clauses

reverse([],A,A).

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

Goal.reverse([1,2,3],[],R)

R= [3,2,1]

1. Write a Prolog program to find check whether a list is a palindrome or not.