CSE 6th SEMESTER

ARTIFICIAL INTELLIGENCE LAB

Lab Code: PCS - 601

L	T	P	C
0	0	2	1

1. Write a prolog program to find the sum of elements in given list.

```
domains
    x = integer
    l = integer*

predicates
    sum(l,x)

clauses
    sum([],0).

    sum([X|List],Sum) :-
        sum(List,Sum1),
        Sum = X + Sum1.
```

Output :

```
Goal: sum([1,2,3,4],Sum)
Sum=10
1 Solution

Goal: sum([-2,-1,1,2],Sum)
Sum=0
1 Solution

Goal: sum([],Sum)
Sum=0
1 Solution

Goal: sum([1],Sum)
Sum=1
1 Solution
```

2. Write a prolog program to delete an element from given list.

```
domains

    list=symbol*


predicates

    del(symbol,list,list)


clauses

    del(X,[X|Tail],Tail).

    del(X,[Y|Tail],[Y|Tail1]):-
        del(X,Tail,Tail1).
```

OUT PUT

```
=======

Goal: del(c,[a,b,c,d,e],NewList)

NewList=["a","b","d","e"]

1 Solution

--------------------------------

Goal: del(a,[b,a,c,a],L)

L=["b","c","a"]

L=["b","a","c"]

2 Solutions
```

3. Write a prolog code to find the last element of a given input list.

```
domains

    list=symbol*


predicates

    last(list)


clauses

    last([X]):-
        write("\nLast element is : "),
        write(X).

    last([Y|Tail]):-
        last(Tail).
```

OUT PUT

```
=======

Goal: last([a,b,c,d,e])

Last element is : e

Yes
```

4. **Write a prolog program to check if given element is a member of a given list.**

```
domains

    list=integer*


predicates

    findnum(integer,list)


clauses

    findnum(X,[]):-
        write("\nNumber Is Not Found").

    findnum(X,[X|Tail]):-
        write("\nNumber Is Found").

    findnum(X,[Y|Tail]):-
        findnum(X,Tail).
```

OUT PUT

```
=======

Goal: findnum(3,[1,2,3,4,5])

Number Is Found

Yes

----------------------------

Goal: findnum(6,[1,2,3,4,5])

Number Is Not Found

Yes

----------------------------

Goal: findnum(2,[1,2,2,1])

Number Is Found

Yes
```

5. **Write a prolog program to append list L1 to list L2 and bind the result to list L3.**

```
includes "domains.pro"
predicates
append(symbol_list, symbol_list, symbol_list)
append(real_list, real_list, real_list)
append(char_list, char_list, char_list)
append(integer_list, integer_list, integer_list)
clauses
append([],L,L)
append([H1|T1],L2,[H1|T3] if append(T1,T2,T3))
```

6. **Write a prolog code to find the maximum element of a list.**

```
domains
    list = integer*
    Max = integer
predicates
    maximum_no(list,integer)
clauses
    maximum_no([],Max):-
        write("Maximum No in List is:: ",Max),nl.
    maximum_no([H|T],Max):-
        H>Max,
        N = H,
        maximum_no(T,N).
    maximum_no(L,Max):-
        maximum_no(L,Max).
```

7. **Write a prolog predicate to reverse the order of the elements of a given input list.**

```
domains
    list=integer*
predicates
    reverse_list(list,list)
    reverse(list,list,list)
clauses

    reverse_list(Inputlist,Outputlist):-
    reverse(Inputlist,[],Outputlist).
    reverse([],Outputlist,Outputlist).

    reverse([Head|Tail],List1,List2):-
        reverse(Tail,[Head|List1],List2).
```

Output :

```
Goal: reverse_list([1,2,
3],X)
X=[3,2,1]
1 Solution
Goal:
```