Indian Institute of Engineering Science & Technology, Shibpur

Department of Computer Science & Technology.

8th Semester Artificial Intelligence Laboratory.

ASSIGNMENT- 2

(Cut, Set, List Processing-II)

Duration- 6 periods. **Full Marks (including Viva Voce)-30**

Write PROLOG programs

1. To add an element to a list provided it is not present in the list.
2. To delete first occurrence of an element from a list.
3. To delete all occurrences of an element from a list.
4. To remove the first occurrence of an element X in L with Y giving the result in L1.
5. has_duplicate(L), that determines whether list L has duplicate elements.
6. To substitute all occurrences of an element by another element in a list.
7. To determine whether a list is a sub list of another list.
   A list is a sub list of another list if it's elements are present in another list consecutively and in the same order.
8. To determine whether an element is a member of a set.
9. To determine whether a set is a subset of another set.
10. To determine intersection of two sets.
11. To determine union of two sets.
12. To determine difference of two sets.
13. To determine symmetric difference of two sets.
14. To delete n^{th} element in L, leaving the rest in L1.
15. To replace n^{th} element by another element X in L, leaving the resultant list in L1.

For the problems 16 – 17 assume L1, L2 and L denote lists of terms.

16. Interleave alternate elements of L1 and L2 into L. For example,
    if L1= [a, b, c] and
    L2= [1, 2], then L= [a, 1, b, 2, c].
17. Transpose L1, L2 into L. That is, if L1= [a, b, c] and L2= [1, 2, 3], then L= [(a, 1), (b, 2), (c, 3)].

For the problems 18 - 27 assume L and L1 is a list of terms.

18. remove_every_other (L, L1) that is true if list L1 is just list L with every other element removed (the two lists should have the same first element).
19. cutlast (L, L1) that defines L1 to be obtained from L with last element removed.
20. trim (N, L, L1) that defines L1 to be obtained from L with first N elements removed.
21. trimlast (N, L, L1) that defines L1 to be obtained from L with last N elements removed.
22. exchange_first_last(L, L1), defines that L1 to be obtained from L with first and last elements exchanged. That is,

```
?-exchange_first_last([a, b, c, d, e], X).
  X= [e, b, c, d, a]
```

23 circular_left_shift(L, L1). That is,
if L= [a, b, c, d, e, f] then
L1= [b, c, d, e, f, a]

24. circular_right_shift(L, L1). That is,
if L= [a, b, c, d, e, f] then
L1= [f, a, b, c, d, e]
[Try using circular_left_shift(L, L1) in 23 to implement circular_right_shift(L, L1).]
25. To delete the middle element from an odd-numbered list L into a list L1.
26. To delete two middle elements from an even-numbered list L into a list L1.
27. To implement unfold (L, L1) that reverses the elements of (an odd numbered) list L, from 1 to middle-1 elements and middle+1 to last element and store the result in L1.