Indian Institute of Engineering Science & Technology, Shibpur

Department of Computer Science & Technology.

8th Semester Artificial Intelligence Laboratory.

ASSIGNMENT- 4

(Accumulator, List Processing-III)

Duration- 6 periods. **Full Marks (including Viva Voce)-20**

Write PROLOG programs

1. *To find factorial N using accumulator.
2. To find length of a list using accumulator.
3. To remove duplicate elements from a list using accumulator.
4. *To remove duplicate elements from a list without using accumulator.
5. *To reverse a list using accumulator.

For the problems 6 - 16 assume L is a list of terms.

6. *has_duplicate(L), that determines whether list L has duplicate elements.
7. *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).
8. *cutlast(L, L1) that defines L1 to be obtained from L with last element removed.
9. *trim(N, L, L1) that defines L1 to be obtained from L with first N elements removed.
10. *trimlast(N, L, L1) defines that L1 to be obtained from L with last N elements removed.
11. *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]
12. *circular_left_shift(L, L1). That is,

    if L= [a, b, c, d, e, f] then

    L1= [b, c, d, e, f, a]
13. *circular_right_shift(L, L1). That is,

    if L= [a, b, c, d, e, f] then

    L1= [f, a, b, c, d, e]

14. *To delete the middle element from an odd-numbered list L into a list L1.
15. *To delete two middle elements from an even-numbered list L into a list L1.
16. *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.

    For the problems 17 – 18 assume L1, L2 and L denote lists of terms.

17. *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].
18. *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)].
19. *Suppose that L1 and L2 are lists of numeric values. Find Inner product (L1, L2, X) that defines X to be inner product of two vectors L1, L2.
20. *Define a predicate to "flatten" a list by constructing a list containing no lists as elements, but containing all of the atoms of the original list. For example, consider the following goal and its corresponding answer.

    ?- flatten ([a, [b, c], [[d], [ ], e]], L).

    {L= [a, b, c, d, e]}

*marked problems are not done in the class.