Indian Institute of Engineering Science & Technology, Shibpur

Department of Computer Science & Technology.

8th Semester Artificial Intelligence Laboratory.

ASSIGNMENT- 8

(Miscellaneous Problems)

Duration – 3 periods. **Full marks (including Viva Voce) – 10**

Write Prolog programs for

*1. **Eight Queens' Problem.**

Eight Queens are to be placed in an 8x8 chess board such that no queen attack each other.

2. **Tower of Hanoi Problem.**

The tower of Hanoi is a game played with three poles and a set of N discs. The discs are graded in diameter, and fit onto the poles by means of a hole cut through the centre of each disc. Initially all the discs are on the left-hand pole. The object of the game is to move all the discs onto the centre pole. The right-hand pole can be used as a "spare" pole, a temporary resting place for discs. Each time a disc is moved from one pole to another, two constraints must be observed: only the top disc on a pole can be moved, and no disc may be placed on top of a smaller one.

3. Implement an Automatic Theorem Prover in Propositional Logic.

Indian Institute of Engineering Science & Technology, Shibpur
Department of Computer Science & Technology.
8th Semester Artificial Intelligence Laboratory.
ASSIGNMENT- 1
(Simple List Processing)

Duration- 3 periods. **Full Marks (including Viva Voce)-10**

Write PROLOG programs

1. To determine whether the first two elements of a list are same.
2. To determine whether a list is *not* a two-element list.
3. To determine whether two lists are of same length.
4. To determine length of a list using your own number system, that does not contain more than two symbols.
5. To determine whether two lists are of same length using the length predicate developed in 4 (previous problem).
6. To find the last element of a list.
7. To find whether an element is a member of a list.
8. To find whether two elements are next to each other in a list.
9. To append two lists in a third list.
10. To find the last element of a list using append predicate developed in 9.
11. To find whether an element is a member of a list using append predicate developed in 9.
12. To find whether two elements are next to each other in a list using append predicate developed in 9.
13. To reverse a list in another list.
14. To determine whether a list is a palindrome [the structure of predicate: palindrome(L)].
15. Write a Prolog program for double(List, ListList), where every element in List appears twice in ListList, e. g., double([1,2,3], [1,1,2,2,3,3]) is true.

Indian Institute of Engineering Science & Technology, Shibpur

Department of Computer Science & Technology.

8th Semester Artificial Intelligence Laboratory.

ASSIGNMENT- 2

(Arithmetic in Prolog)

Duration- 2 periods. **Full Marks (including Viva Voce)-10**

Write Prolog programs

1. To find the **sum** of all elements of a list.
2. To find the **length** of a list.
3. *To find the **average** of all elements of a list using **sum** and **length** defined in Problem 1 and 2.
4. To find the **maximum** number from a list.
5. *To find **gcd** of two integers.
6. *To **generate** all integers **between** two integers N1 and N2, both N1 and N2 included and N2>N1.
7. *To **count** numbers **greater** than 100.0 in a list.
8. *To **split** a list of numbers in two lists such that one contains negative numbers and other contains positive numbers.

For the Problems 9 – 13, you can **only** assume increment by 1 (i. e., +1), decrement by 1 (i. e., -1) and any other predicate defined earlier in 9 – 12.

9. *To recursively **add** two integers.
10. *To **subtract** two integers recursively.
11. * To **multiply** two integers recursively.
12. *To **divide** two integers recursively and get the quotient.
13. *To find the **square root** of an integer.
14. * To find **N!**.
15. *To generate first N **Fibonacci** numbers.

*marked problems are not done in the class.

Indian Institute of Engineering Science & Technology, Shibpur

Department of Computer Science & Technology.

8th Semester Artificial Intelligence Laboratory.

ASSIGNMENT- 3

(Cut, List Processing-II, Set)

Duration- 4 periods. **Full Marks (including Viva Voce)-10**

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. *To substitute all occurrences of an element by another element in a list.
6. 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.
7. To determine whether an element is a member of a set.
8. *To determine whether a set is a subset of another set.
9. To determine intersection of two sets.
10. To determine union of two sets.
11. To determine difference of two sets.
12. *To determine symmetric difference of two sets.
13. *To delete n^{th} element in L, leaving the rest in L1.
14. *To replace n^{th} element by another element X in L, leaving the resultant list in L1.

*marked programs are not done in the class.

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.

Indian Institute of Engineering Science & Technology, Shibpur

Department of Computer Science & Technology.

8th Semester Artificial Intelligence Laboratory.

ASSIGNMENT- 5

(Sorting in Prolog)

Duration- 4 periods. **Full Marks (including Viva Voce)-15**

Implement the following sorting in Prolog.

1. *Bubble Sort.
2. Selection Sort.
3. Permutation Sort.
4. Insertion Sort.
5. Merge Sort.
6. Quick Sort using Accumulator.
7. *Quick Sort without using Accumulator.
8. *Heap Sort.

*marked assignments are not done in the class.

Indian Institute of Engineering Science & Technology, Shibpur

Department of Computer Science & Technology.

8th Semester Artificial Intelligence Laboratory.

ASSIGNMENT- 6

(Trees in Prolog)

Duration- 5 periods. **Full Marks (including Viva Voce)-15**

Write Prolog programs

1. *To determine whether an element is a member of a binary tree.
2. *To determine whether an element is a member of a binary search tree.
3. *To determine whether an object is a binary tree.
4. To find whether an object is a binary search tree.
5. *To find maximum element from a binary search tree.
6. To find the height of a binary tree.
7. To find the preorder traversal of a binary tree, storing the result in a list.
8. *To find the inorder traversal of a binary tree, storing the result in a list.
9. To find the postorder traversal of a binary tree, storing the result in a list.
10. *To insert an element in a binary search tree.
11. *To delete a leaf node from a binary search tree.
12. To delete a node from a binary search tree.
13. To sort an unordered list into an ordered list using a binary search tree and inorder traversal.
14. To sort an unordered list into an ordered list using insertions in a binary search tree and subsequent deletions of minimum elements.
15. Given preorder and inorder traversals of a binary tree in two lists, obtain its postorder traversal in another list.
16. Given postorder and inorder traversals of a binary tree in two lists, obtain its preorder traversal in another list.

*marked problems are done in the class.

Indian Institute of Engineering Science & Technology, Shibpur

Department of Computer Science & Technology.

8th Semester Artificial Intelligence Laboratory.

ASSIGNMENT- 7

(Production Systems in Prolog)

Duration- 6 periods. **Full Marks (including Viva Voce)-15**

1. ***Monkey Banana Problem.**

There is a monkey at the door into a room. In the middle of the room a banana is hanging from the ceiling. The monkey is hungry and wants to get the banana, but he cannot stretch high enough from the floor. At the window of the room there is a box that the monkey can use. The monkey can perform the following actions: walk on the floor, climb the box, push the box around (if he is already at it), and grasp the banana if he is standing on the box and directly underneath the banana.

Can the monkey grasp the banana?

Answer the question by writing a Prolog Program.

2. **Water Jug Problem.**

There are two water jugs with capacities of 4 litres and 3 litres respectively. Neither have any measuring markers on them. The water jugs are both initially empty. One can fill the water jugs with water completely or partially. One can empty completely or partially filled jugs both completely or partially. One can transfer the content (water) of one jug (partially or completely} to another jug. **Write Prolog program that outputs the steps to get exactly 2 litres of water in 4 litres jug.**

3. Missionaries and Cannibals Problem.

Three missionaries and three cannibals must cross a river using a boat which can carry at most two people, under the constraint that, for both banks, if there are missionaries present on the bank, they cannot be outnumbered by cannibals (if they were, the cannibals would eat the missionaries). The boat cannot cross the river by itself with no people on board. **Write a Prolog program to write the steps to illustrate how can missionaries and cannibals can cross the river?**

4. Tiger, Goat and Cabbage Problem.

A man is traveling with a tiger, a goat, and a cabbage. We have no idea why he would be travelling with such a strange assortment, but there he is anyway. At one point in his journey he comes to a river which is too deep to wade across, and too wide to swim across so he is in a quandary on how to continue. He notices a small boat tied to the near shore, but the boat is too small to fit all his belongings into, but it is large enough so that he can safely row across with one belonging at a time.

The problem is that if he rows across with the tiger first, then the goat will eat the cabbage, and if he rows across with the cabbage first, the tiger will eat the goat.

How can he safely cross the river with all his things intact? Write a Prolog program to illustrate the steps.

*****marked problem discussed in class.