Logic Programming – Laboratory 6

Revision

Isabela Drămnesc

1 Exercises

1. Write an efficient program (with accumulators) in Prolog which generates a list of a certain length and its elements are random numbers.

Examples:
$? - generate\_elem\_list(10000000, 2, L).$
will return the list $L = [0, 0, 1, 0, 0, 1, 1, 0, 1, 0|.....]$ which has the length 10000000 and which contains binary elements.
$? - generate\_elem\_list(10, 8, L).$ will return $L = [5, 3, 0, 6, 6, 6, 5, 3, 3|...]$.

2. Write a program which will return the list of elements on the odd positions from a list of integers. Use accumulators.

Examples:
$? - odd\_elem\_list([1, 3, 4, 2, 3, 5], OddElem).$
The result will be $OddElem = [1, 4, 3]$.
$? - odd\_elem\_list([1], OddElem).$
The result will be $L = [1]$.

3. Write a program which will return the list of elements on the even positions from a list of integers. Use accumulators.

Examples:
$? - even\_elem\_list([1, 3, 4, 2, 3, 5], EvenElem).$
The result will be $EvenElem = [3, 2, 5]$.
$? - even\_elem\_list([3], EvenElem).$
The result will be $EvenElem = []$.

4. Write a program which will merge two lists of integers. Use accumulators.

Examples:
$? - merge\_lists([1, 2, 2, 3, 5], [2, 4, 7, 9, 11, 23], Result).$
The result will be $Result = [1, 2, 2, 2, 3, 4, 5, 7, 9, 11, 23]$.

5. Implement the merge-sort algorithm for integers in Prolog – informally it can be formulated as follows: Given a list, divide the list into two halves. Sort the halves and merge the two sorted lists. Sort the list (of the length 10000000) generated at exercise 1.

6. Write a program which returns all the prefixes of a list in 3 ways.

1. The recursive version;
2. Using accumulators;
3. Using open lists/difference lists.

Examples:
1. ?- prefix(L,[1,2,3,f,r,4]).
L = [] ;
L = [1] ;
L = [1, 2] ;
L = [1, 2, 3] ;
L = [1, 2, 3, f] ;
L = [1, 2, 3, f, r] ;
L = [1, 2, 3, f, r, 4] ;

Similar for prefix2 and prefix3.