

Functional and logic programming

- written exam -

Important:

1. Subjects are graded as follows: of - 1p; A – 1.5p; B - 2.5p; C - 2.5p; D - 2.5p.
2. Prolog problems will be resolved using SWI Prolog. The following are required: (1) explanation of the code and of the reasoning behind it; (2) recursive model that solves the problem, for all the predicates used; (3) specification of every predicate (parameters and their meaning, flow model, type of the predicate - deterministic/non-deterministic).
3. Lisp problems will be resolved using Common Lisp. The following are required: (1) explanation of the code and of the reasoning behind it; (2) recursive model that solves the problem, for each function used; (3) specification of every function (parameters and their meaning).

A. Let L be a list of numbers and given the following PROLOG predicate definition **f(list, integer)**, with the flow model (i, o):

f([], 0).

f([H|T], S):-**f(T, S1)**, S1<H,!, S is H.

f(_|T], S):-**f(T, S1)**, S is S1.

Rewrite the definition in order to avoid the recursive call **f(T, S)** in both clauses. Do NOT redefine the predicate. Justify your answer.

B. Given a nonlinear list that contains numerical and non-numerical atoms, write a LISP program that builds a list that has the same structure as the initial list, but each (sub)list contains a single element on the superficial level, and that element is computed as the difference between the maximum and the minimum values on the superficial level of the corresponding (sub)list in the initial list. It is guaranteed that each (sub)list contains at least a numerical atom. For example, for the list (F A 12 13 (B 11 (A D 15) C 3 C (1 F 6) 1) 18 11 D (A 7 F 9) F)) the result will be (7 (10 (0) (5)) (2)) (7 is the difference between 18 and 11, 10 is the difference between 11 and 1, 0 is the difference between 15 and 15, etc.).

C. Given a list made of integer numbers, generate using PROLOG the list of arrangements with even number of elements, having the sum an odd number. Write mathematical models and flow models for the predicates used. For example, for the list $L=[2,3,4] \Rightarrow [[2,3],[3,2],[3,4],[4,3]]$ (not necessarily in this order).

D. An n-ary tree is represented in Lisp as (node subtree1 subtree2 ...). Write a function to replace all nodes on odd levels with a given value **e**. The root level is assumed zero. **A MAP function shall be used.**

Example for the tree (a (b (g)) (c (d (e)) (f))) and **e=h** => (a (h (g)) (h (d (h)) (h)))