

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. Given the following PROLOG predicate definition **f(integer, integer)**, with the flow model (i, o):

f(100, 1):-!.

f(K,X):-K1 is K+1, **f(K1,Y)**, Y>1, !, K2 is K1-1, X is K2+Y.

f(K,X):-K1 is K+1, **f(K1,Y)**, Y>0.5, !, X is Y.

f(K,X):-K1 is K+1, **f(K1,Y)**, X is Y-K1.

Rewrite the definition in order to avoid the recursive call **f(J,V)** in all 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 counts for how many sublists (including the initial list) the total number of numerical atoms on odd levels is equal with the total number of non-numerical atoms on odd levels. The superficial level is odd. For example, for the list (A B 12 (5 D (A F (B) D (5 F) 1) 5) C 9 (F 4 (D) 9 (F (H 7) K) (P 4)) X) the result will be 4 (the counted lists being (5 F) (H 7) (P 4) (5 D (A F (B) D (5 F) 1) 5)).

C. Write a PROLOGO program that generates the list of arrangements of k elements from a list of integer numbers, having the given product P . Write the mathematical models and flow models for the predicates used. For example, for the list $[2, 5, 3, 4, 10]$, $k=2$ and $P=20 \Rightarrow [[2,10],[10,2],[5,4],[4,5]]$ (not necessarily in this order).

D. An n-ary tree is represented in Lisp as (node subtree1 subtree2 ...). Write a Lisp function to verify whether a node **x** occurs on an even level of the tree. The root level is assumed zero. **A MAP function shall be used.**

Example for the tree (a (b (g)) (c (d (e)) (f)))

a) $x=g \Rightarrow T$

b) $x=h \Rightarrow \text{NIL}$