

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([], -1).

f([H|T],S):-H>0, **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. An n-ary tree can be stored as a linear list in which each node is followed by its number of children. Given a linear list that represents an n-ary tree, write a LISP program that determines, in the form of a list, the k-th descendent of the root from the tree. For example, for the tree (A 5 B 2 E 0 F 3 G 0 H 0 I 0 C 1 J 1 K 2 L 0 M 0 D 4 N 0 O 0 P 2 R 0 S 1 T 0 Q 0 U 0 V 1 Z 2 T 0 W 0) and k = 3 the result will be (D 4 N 0 O 0 P 2 R 0 S 1 T 0 Q 0), and for the same tree and k = 5 the result will be (V1 Z 2 T 0 W 0).

C. Write a PROLOG program that generates the list of all subsets of k elements (all elements being odd numbers) in arithmetic progression. Write the mathematical models and flow models for the predicates used. For example, for $L=[1,5,2,9,3]$ and $k=3 \Rightarrow [[1,5,9],[1,3,5]]$ (not necessarily in this order).

D. An n-ary tree is represented in Lisp as (node subtree1 subtree2 ...). Write a Lisp function to determine the number of nodes on level **k**. The root level is assumed zero. **A MAP function shall be used.** ***Example*** for the tree (a (b (g)) (c (d (e)) (f)))

a) k=2 => nr=3 (g d f) **b)** k=4 => nr=0 ()