

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(20, -1):-!.

f(I,Y):-J is I+1, **f(J,V)**, V>0, !, K is J, Y is K.

f(I,Y):-J is I+1, **f(J,V)**, Y is V-1.

Rewrite the definition in order to avoid the recursive call **f(J,V)** 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 permutations of the set $1..N$, having the property that the absolute value of the difference between 2 consecutive values from the permutation is ≥ 2 . Write the mathematical models and flow models for the predicates used. For example, for $N=4 \Rightarrow [[3,1,4,2], [2,4,1,3]]$ (not necessarily in this order).

D. Given a nonlinear list, write a Lisp function to return the list with all even numerical atoms from an odd level removed. The superficial level is assumed 1. **A MAP function shall be used.**

Example **a)** if the list is (1 (2 A (4 A)) (6)) => (1 (2 A (A)) (6))

b) if the list is (1 (2 (C))) => (1 (2 (C)))