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

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 containing both numerical and non-numerical atoms, write a LISP program that builds a list that contains on even levels the greatest numerical atom, and for odd levels the lowest numerical atom (we assume that each level of the list contains at least a numerical atom), but in reverse order (so the minimum on level 1 is the last element, the maximum on level 2 is the penultimate element, etc.). For example, for the list (A B 12 (5 D (A F (10 B) D (5 F) 1)) C 9 (F 4 (D) 9 (F (H 7) K) (P 4)) X) the result will be (10 1 9 9). You are not allowed to use function *reverse* from Lisp.

C. Given a list made of integer numbers, generate in PROLOG the list of all subsets with even number of elements. Write the mathematical models and flow models for the predicates used. For example, for the list L=[2,3,4] ⇒ [[],[2,3],[2,4],[3,4]] (not necessarily in this order).

D. An n-ary tree is represented in Lisp as (node subtree1 subtree2 ...). Write a Lisp program to return the ***height*** of a node of a tree. **A MAP function shall be used.**
Example for the tree (a (b (g)) (c (d (e)) (f)))
a) nod=e => the height is 0 **b)** nod=v => the height is -1 c**)** nod=c => the height is 2.