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 containing both numerical and non-numerical atoms, write a LISP program that computes the greatest common divisor of odd numbers on even levels of the list. The superficial level is considered to be 1. For example, for the list (A B 12 (9 D (A F (75 B) D (45 F) 1) 15) C 9), the result will be 3. We assume there is at least one odd number at each even level of the list. You are not allowed to use the predefined Lisp function *gcd*.

C. Write a PROLGO 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 ⇒ [[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 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.