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. Given a nonlinear list of both numerical and non-numerical atoms, write a LISP program that builds a linear list composed only from those non-numerical atoms that occur an even number of times in the initial list. The result will contain each element only once, in reverse order of the initial list. **For example**, for the list (F A 2 3 (B 1 (A D 5) C C (F)) 8 11 D (A F) F), the result will be (C D F). You are NOT allowed to use predefined LISP functions *reverse* or *member*.

C. Write a PROLOG program that generates the list of all subsets of k elements 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 ⇒ [[1,2,3],[1,5,9],[1,3,5]] (not necessarily in this order).

D. Given a nonlinear list, write a Lisp function to return the list with all atoms on the level **k** replaced by 0. The superficial level is assumed 1. **A MAP function shall be used.**
Example for the list (a (1 (2 b)) (c (d))) **a)** k=2 => (a (0 (2 b)) (0 (d)))
b) k=1 => (0 (1 (2 b)) (c (d))) **c)** k=4 => the list does not change