Functional and logic programming
- **written exam -**

Important:

1. Subjects are graded as follows: By default - 1p; A – 2p; B - 4p; C - 3p.
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),H<S1,!,S is H+S1.
f([_|T],S):-f(T,S1), S is S1+2.
```

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. 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).

C. Given a nonlinear list, write a Lisp function to replace all even numerical values with their natural successor. **A MAP function shall be used.**
Example for the list (1 s 4 (2 f (7))) the result is (1 s 5 (3 f (7))).