

## 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)**,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.** Given a nonlinear list that contains numerical and non-numerical atoms, write a Lisp program that builds a list that has a level for each level of the initial list and on each level has three elements: the number of numerical atoms on that level from the initial list, a sublist that contains these information for the rest of the levels and the numbers of nonnumerical atoms from that level in the initial list. For example, for the list (A B (4 A 3) 11 (5 (A (B) C 10) (1(2(3(4)5)6)7) X Y Z)) the result will be (1 (3 (3 (2 (2 (1 0) 0) 1) 2) 4) 2).

**C.** Write a PROLOG program that generates the list of all arrangements of  $k$  elements with the value of sum of all elements from each arrangement equal with a given  $S$ , from a list of integers. Write the mathematical models and flow models for the predicates used. For example, for the list  $[6, 5, 3, 4]$ ,  $k=2$  and  $S=9 \Rightarrow [[6,3],[3,6],[5,4],[4,5]]$  (not necessarily in this order).

**D.** Given a nonlinear list, write a Lisp function to replace all the odd values from even levels with their natural successor. The superficial level is assumed 1. **A MAP function shall be used.**  
**Example** for the list (1 s 4 (3 f (7))) the result is (1 s 4 (4 f (7))).