

## 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.** Given the following PROLOG predicate definition **f(integer, integer)**, with the flow model (i, o):

f(50, 1):-!.

f(I,Y):-J is I+1, **f(J,S)**, S<1, !, K is I-2, Y is K.

f(I,Y):-J is I+1, **f(J,Y)**.

Rewrite the definition in order to avoid the recursive call **f(J,V)** in both clauses. Do NOT redefine the predicate. Justify your answer.

**B.** Given a list composed of integer numbers, generate in PROLOG the list of arrangements of N elements ending with an odd value and have the sum S given. Write the mathematical models and flow models for the predicates used. For example, for the list  $L=[2,7,4,5,3]$ ,  $N=2$  and  $S=7 \Rightarrow [[2,5], [4,3]]$  (not necessarily in this order).

**C.** Write a Lisp function to substitute all numerical values at any level of a given nonlinear list with a given value **e**. **A MAP function shall be used.**

**Example**, for the list (1 d (2 f (3))), **e**=0 the result is (0 d (0 f (0))).