

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. The following function definition in LISP is given

```
(DEFUN F(L)
  (COND
    ((NULL L) NIL)
    (> (F (CAR L)) 0) (CONS (F (CAR L)) (F (CDR L))))
    (T (F (CAR L)))
  )
)
```

Rewrite the definition in order to avoid the repeated recursive call **(F (CAR L))**. Do NOT redefine the function. Do NOT use SET, SETQ, SETF. Justify your answer.

B. Given a heterogeneous list formed by numbers and non-empty linear numerical lists, write a SWI-Prolog program that computes the difference between the smallest maximum number from the sublists and the greatest minimum value from the sublists. We assume the input list contains at least one sublist. For example, for the list `[[4, 2, 18], 7, 2, -3, [6, 9, 11, 3], 4, [5, 9, 19]]` the result will be 6.

C. Write a PROLOG program that generates the list of all subsets of sum **S** given, using the elements of a list, such that the number of even elements from each subset is even. Write the mathematical models and flow models for the predicates used. For example for the list [1, 2, 3, 4, 5, 6, 10] and $S=10 \Rightarrow [[1,2,3,4], [4,6]]$.

D. Given a nonlinear list, write a Lisp function to return the list with all the numerical atoms that are multiple of 3 removed. **A MAP function shall be used.**

Example **a)** if the list is (1 (2 A (3 A)) (6)) => (1 (2 A (A)) NIL)

b) if the list is (1 (2 (C))) => (1 (2 (C)))