

## 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.** The following function definition in LISP is given

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

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

**B.** Write a PROLOG program that generates the list of all subsets with N elements, using the elements of a list, such that the sum of elements from a subset is an even number. Write the mathematical models and flow models for the predicates used. For example, for the list  $L=[1, 3, 4, 2]$  and  $N=2 \Rightarrow [[1,3], [2,4]]$ .

**C.** An n-ary tree is represented in Lisp as ( node subtree1 subtree2 ...). Write a Lisp program to return the ***height*** of a node of a tree. **A MAP function shall be used.**

**Example** for the tree (a (b (g)) (c (d (e)) (f)))

**a)** nod=e => the height is 0      **b)** nod=v => the height is -1      **c)** nod=c => the height is 2.