

## 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(L)
  (COND
    ((NULL L) NIL)
    ((LISTP (CAR L)) (APPEND (F (CAR L)) (F (CDR L)) (CAR (F (CAR L)))))
    (T (LIST(CAR L)))
  )
)
```

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

**B.** For a given value  $N$ , generate the list of all permutations with elements  $N, N+1, \dots, 2*N-1$  with the property that the absolute value between two consecutive values from the permutation is  $\leq 2$ . Write the mathematical models and flow models for the predicates used.

**C.** An n-ary tree is represented in Lisp as ( node subtree1 subtree2 ...).. Write a function to return the list of nodes on even levels, in increasing level order (0, 2, ...). The root level is assumed zero. **A MAP function shall be used.**

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