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(L1 L2)
        (APPEND (F (CAR L1) L2)
                (COND
                    ((NULL L1) (CDR L2))
                    (T (LIST (F (CAR L1) L2) (CAR L2)))
                )
        )
)
```

Rewrite the definition in order to avoid the double recursive call **(F (CAR L1) L2)**. Do NOT redefine the function. Do NOT use SET, SETQ, SETF. Justify your answer.0

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

C. Given a nonlinear list, write a Lisp function to replace the numerical values on off levels and greater than a given value k to their natural predecessor. The superficial level is assumed 1. **A MAP function shall be used.** ***Example*** for the list (1 s 4 (3 f (7))) and

a) k=0 the result is (0 s 3 (3 f (6))) **b)** k=8 the result is (1 s 4 (3 f (7)))