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(N)
        (COND
                ((= N 1) 1)
                (> (F (- N 1)) 2) (- N 2))
                (> (F (- N 1)) 1) (F (- N 1)))
                (T (- (F (- N 1)) 1))
        )
)
```

Rewrite the definition in order to avoid the repeated call **(F (- N 1))**. Do NOT redefine the function. Do NOT use SET, SETQ, SETF. Justify your answer.

B. Given a numeric linear list, write a SWI-PROLOG program that returns (as a list of pairs) all possible partitions of the initial list in two sublists, such that the average value of the elements from the first sublist is smaller or equal to the greatest common divisor of the elements from the second sublist. **For example**, for the list [8, 4, 6, 1], the result will be (not necessarily in this order): [[[1, 4, 8], [6]], [[1, 6, 4], [8]], [[1, 6], [4, 8]], [[1], [6, 4, 8]]].

C. Write a PROLOG program that generates the list of all subsets with at least N elements such that the value of sum of all elements from each subset is divisible with 3, from a list of integers. Write the mathematical models and flow models for the predicates used. For example, for the list L=[2,3,4] and **N**=1 ⇒ [[3],[2,4],[2,3,4]] (not necessarily in this order).

D. An n-ary tree is represented in Lisp as (node subtree1 subtree2 ...). Write a Lisp function to determine the number of nodes on level **k**. The root level is assumed zero. **A MAP function shall be used.** ***Example*** for the tree (a (b (g)) (c (d (e)) (f)))
a) k=2 => nr=3 (g d f) **b)** k=4 => nr=0 ()