

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 0) 0)
    (> (F (- N 1)) 1) (- N 2))
    (T (+ (F (- N 1)) 1))
  )
)
```

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

B. Given a nonlinear list composed of numbers greater or equal to 2, write a SWI-PROLOG program that replaces each nonprime number with the sum of its own proper divisors. Repeat the process until the list contains only prime numbers. **For example**, for the list [10, 20, 30, 40] the result will be [7, 7, 41, 7] (the initial list becomes first [7, 21, 41, 49], then [7, 10, 41, 7] and finally [7, 7, 41, 7]). Return only the final list.

C. Write a PROLOG program that generates the list of all permutations with the property the absolute value of difference between two consecutive values from each permutation is ≤ 3 . Write the mathematical models and flow models for the predicates used. For example, for $L=[2,7,5] \Rightarrow [[2,5,7], [7,5,2]]$ (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 path from the root to a given node. **A MAP function shall be used.**

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

(a) nod=e => (a c d e) **(b)** nod=v => ()