

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(L)
  (COND
    ((ATOM L) -1)
    ((> (F (CAR L)) 0) (+ (CAR L) (F (CAR L)) (F (CDR L))))
    (T (F (CDR 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. 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 subsets with value of sum for each subset odd number and also odd numbers of odd values from each subset. Write the mathematical models and flow models for the predicates used. For example, for $[2,3,4] \Rightarrow [[2,3],[3,4],[2,3,4]]$ not necessarily in this order).

D. Given a nonlinear list, write a Lisp function to return the list with all occurrences of the element **e** replaced by the value **e1**. **A MAP function shall be used.**

Example **a)** if the list is (1 (2 A (3 A)) (A)), **e** is A and **e1** is B => (1 (2 B (3 B)) (B))

b) if the list is (1 (2 (3))) and **e** is A => (1 (2 (3)))