

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. Given the following PROLOG predicate definition **f(list, integer)**, with the flow model (i, o):

`f([], -1):-!`

`f([_|T], Rez):- f(T,S), S<1, !, Y is S+2.`

`f([H|T], Rez):- f(T,S), S<0, !, Y is S+H.`

`f([_|T], Rez):- f(T,S), Y is S.`

Rewrite the definition in order to avoid the recursive call **f(T,S)** in all clauses. Do NOT redefine the predicate. Justify your answer.

B. Write a PROLOG program that generates the list of all subsets with values between the $[a, b]$ interval such that the sum of elements from each subset is an odd value. Write the mathematical models and flow models for the predicates used. For example, for $a=2$ and $b=4 \Rightarrow [[2,3],[3,4],[2,3,4]]$ (not necessarily in this order).

C. Write a Lisp function to substitute all numerical values at any level of a given nonlinear list with a given value **e**. **A MAP function shall be used.**

Example, for the list (1 d (2 f (3))), **e**=0 the result is (0 d (0 f (0))).