

# Programare logică și funcțională

## - examen scris -

### Notă

1. Subiectele se notează astfel: of - 1p; A - 1.5p; B - 2.5p; C - 2.5p; D - 2.5p.
2. Problemele Prolog vor fi rezolvate în SWI Prolog. Se cere: (1) explicarea codului și a raționamentului; (2) modelul recursiv de rezolvare, pentru fiecare predicat folosit; (3) specificarea fiecărui predicat (semnificația parametrilor, model de flux, tipul predicatului - determinist/nedeterminist).
4. Problemele Lisp vor fi rezolvate în Common Lisp. Se cere: (1) explicarea codului și a raționamentului; (2) modelul recursiv de rezolvare, pentru fiecare funcție folosită; (3) specificarea fiecărei funcții (semnificația parametrilor).

A. Fie L o listă numerică și următoarea definiție de predicat PROLOG **f(list, integer)**, având modelul de flux (i, o):

$f([], -1)$ .

$f([H|T], S) :- H > 0, \text{f}(T, S1), S1 < H, !, S \text{ is } H$ .

$f([_|T], S) :- \text{f}(T, S1), S \text{ is } S1$ .

Rescrieți această definiție pentru a evita apelul recursiv **f(T,S)** în ambele clauze. Nu redefiniți predicatul. Justificați răspunsul.

- B.** Dându-se o listă neliniară care conține atomi numerici și nenummerici, se cere un program Lisp care construiește o listă care are câte un nivel pentru fiecare nivel existent în lista inițială și pe fiecare nivel are 3 elemente: numărul atomilor numerici de pe acest nivel din lista inițială, o sublistă care conține aceste informații pentru restul nivelurilor și numărul atomilor nenummerici de pe acest nivel din lista inițială. De exemplu, pentru lista (A B (4 A 3) 11 (5 (A (B) C 10) (1(2(3(4)5)6)7) X Y Z) rezultatul va fi (1 (3 (3 (2 (2 (1 0) 0) 1) 2) 4) 2).

- C. Dându-se o listă formată din numere întregi, să se genereze în PROLOG lista submulțimilor cu cel puțin **N** elemente având suma divizibilă cu 3. Se vor scrie modelele matematice și modelele de flux pentru predicatele folosite.

**Exemplu**- pentru lista  $L=[2,3,4]$  și  $N=1 \Rightarrow [[3],[2,4],[2,3,4]]$  (nu neapărat în această ordine)

D. Un arbore n-ar se reprezintă în LISP astfel ( nod subarbore1 subarbore2 .....)

Se cere să se înlocuiască nodurile de pe nivelul **k** din arbore cu o valoare **e** dată. Nivelul rădăcinii se consideră a fi 0.

**Se va folosi o funcție MAP.**

**Exemplu** pentru arborele (a (b (g)) (c (d (e)) (f))) și **e=h**

- a)  $k=2 \Rightarrow (a (b (h)) (c (h (e)) (h)))$
- b)  $k=4 \Rightarrow (a (b (g)) (c (d (e)) (f)))$