

# Programare logică și funcțională

## - examen scris -

### Notă

1. Subiectele se notează astfel: of - 1p; A – 2p; B - 4p; C - 3p.
2. Problema Prolog (B) vor fi rezolvată 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).
3. Problema Lisp (C) va fi rezolvată 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):

$f([], -1)$ .

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

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

Rescrieți această definiție pentru a evita apelul recursiv **f(T,S)** în ambele clauze, fără a redefini logica clauzelor. Justificați răspunsul.

**B.** Să se scrie un program PROLOG care generează lista submulțimilor cu **N** elemente, cu elementele unei liste, astfel încât suma elementelor dintr-o submulțime să fie număr par. Se vor scrie modelele matematice și modelele de flux pentru predicatele folosite. **Exemplu**- pentru lista  $L=[1, 3, 4, 2]$  și  $N=2 \Rightarrow [[1,3], [2,4]]$

**C.** Se consideră o listă neliniară. Să se scrie o funcție LISP care să aibă ca rezultat lista inițială din care au fost eliminați toți atomii numerici pari situați pe un nivel impar. Nivelul superficial se consideră a fi 1. **Se va folosi o funcție MAP.**

**Exemplu**

**a)** dacă lista este (1 (2 A (4 A)) (6)) => (1 (2 A (A)) (6))

**b)** dacă lista este (1 (2 (C))) => (1 (2 (C)))