

# 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).
3. 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 având modelul de flux (i, o):

$f([], -1)$ .

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

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

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 conținând atât atomi numerici, cât și nenumeriți, se cere un program LISP care să construiască o listă liniară formată doar din acei atomi nenumeriți care apar de un număr par de ori în lista inițială. Rezultatul va conține fiecare element o singură dată, în ordine inversă față de ordinea în care elementele apar în lista inițială. **De exemplu,** pentru lista (F A 2 3 (B 1 (A D 5) C C (F)) 8 11 D (A F) F), rezultatul va fi (C D F). NU se pot folosi funcțiile predefinite *reverse* sau *member* din Lisp.

**C.** Dându-se o listă formată din numere întregi, să se genereze în PROLOG lista aranjamentelor cu număr par de elemente, având suma număr impar. Se vor scrie modelele matematice și modelele de flux pentru predicatele folosite.

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

- D. Se dă o listă neliniară și se cere înlocuirea valorilor numerice care sunt mai mari decât o valoare **k** dată și sunt situate pe un nivel impar, cu numărul natural predecesor. Nivelul superficial se consideră 1. **Se va folosi o funcție MAP.**

**Exemplu** pentru lista (1 s 4 (3 f (7))) și

a)  $k=0$  va rezulta (0 s 3 (3 f (6)))

b)  $k=8$  va rezulta (1 s 4 (3 f (7)))