

# 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):

$f([], -1).$

$f([H|T], S) :- f(T, S1), S1 < 1, S \text{ is } S1 - H, !.$

$f([_|T], S) :- f(T, S).$

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 verifice dacă atomii numerici din listă formează o secvență de valori în ordine crescătoare. **De exemplu**, pentru lista `(A B 1 (2 C D) 3 4 (F T 6 10 (A E D) (34) F) 111))` rezultatul este **adevărat** (T), iar pentru lista `(A B 1 (2 C D) 3 4 (F T 6 1 (A E D) (34) F) 111))` rezultatul este **fals** (NIL).

**C.** Să se scrie un program PROLOG care generează lista submulțimilor de sumă **S** dată, cu elementele unei liste, astfel încât numărul elementelor pare din submulțime să fie par. **Exemplu** pentru lista [1, 2, 3, 4, 5, 6, 10] și **S**=10  $\Rightarrow$  [[1,2,3,4], [4,6]].

- D. Un arbore n-ar se reprezintă în LISP astfel (nod subarbore1 subarbore2 .....). Se cere să se determine numărul de noduri de pe nivelul **k**. Nivelul rădăcinii se consideră 0. **Se va folosi o funcție MAP.**

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

**a)**  $k=2 \Rightarrow nr=3$  (g d f)    **b)**  $k=4 \Rightarrow nr=0$  ()