

# 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 următoarea definiție de predicat PROLOG **f(integer, integer)**, având modelul de flux (i, o):

$f(1, 1):-!$ .

$f(K,X):-K1 \text{ is } K-1, \text{ f(K1,Y), } Y>1, !, K2 \text{ is } K1-1, X \text{ is } K2.$

$f(K,X):-K1 \text{ is } K-1, \text{ f(K1,Y), } Y>0.5, !, X \text{ is } Y.$

$f(K,X):-K1 \text{ is } K-1, \text{ f(K1,Y), } X \text{ is } Y-1.$

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

**B.** Dându-se o listă formată din numere întregi, să se genereze în PROLOG lista permutărilor având proprietatea că valoarea absolută a diferenței dintre două valori consecutive din permutare este  $\leq 3$ . Se vor scrie modelele matematice și modelele de flux pentru predicatele folosite.

**Exemplu**- pentru lista  $L=[2,7,5] \Rightarrow [[2,5,7], [7,5,2]]$  (nu neapărat în această ordine)

**C.** Un arbore n-ar se reprezintă în LISP astfel (nod subarbore1 subarbore2 .....). Se cere să se verifice dacă un nod **x** apare pe un nivel par în arbore. Nivelul rădăcinii se consideră a fi 0. **Se va folosi o funcție MAP.**

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

**a)**  $x=g \Rightarrow T$       **b)**  $x=h \Rightarrow NIL$