

# 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 următoarea definiție de funcție în LISP

```
(DEFUN F(L1 L2)
  (APPEND (F (CAR L1) L2)
    (COND
      ((NULL L1) (CDR L2))
      (T (LIST (F (CAR L1) L2) (CAR L2)))
    )
  )
)
```

Rescrieți această definiție pentru a evita dublul apel recursiv **(F (CAR L1) L2)**. Nu redefiniți funcția. Nu folosiți SET, SETQ, SETF. Justificați răspunsul.

- B.** Dându-se o listă eterogenă formată din numere și liste liniare nevide de numere, se cere un program SWI-PROLOG care să construiască o listă cu valorile minime din acele subliste având suma elementelor număr prim. Lista rezultată va conține elementele în ordine inversă față de ordinea în care sublistele apăreau în lista inițială. **De exemplu**, pentru lista `[[4, 1, 18], 7, 2, -3, [6, 9, 11, 3], 4, [5, 9, 19]]`, rezultatul va fi `[3, 1]`.

- C. Să se scrie un program PROLOG care generează lista aranjamentelor de **k** elemente dintr-o listă de numere întregi, pentru care produsul elementelor e mai mic decât o valoare **V** dată. Se vor scrie modelele matematice și modelele de flux pentru predicatele folosite.

**Exemplu**- pentru lista [1, 2, 3], **k**=2 și **V**=7  $\Rightarrow$  [[1,2],[2,1],[1,3],[3,1],[2,3],[3,2]] (nu neapărat în această ordine)

- D. 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)**  $x=h \Rightarrow NIL$