Programare logică și funcțională - examen scris -

<u>Notă</u>

- 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 LISP

(DEFUN F(G L)

(COND

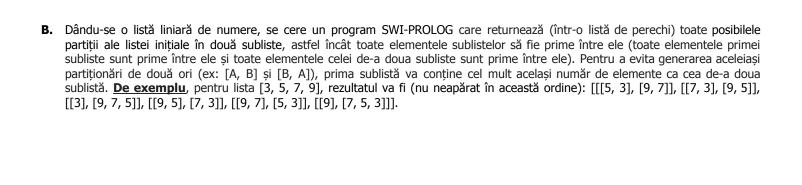
((NULL L) NIL)

(> (FUNCALL G L) 0) (CONS (FUNCALL G L) (F (CDR L))))

(T (FUNCALL G L))

)
```

Rescrieți această definiție pentru a evita apelul repetat (FUNCALL G L). Nu redefiniți funcția. Nu folosiți SET, SETQ, SETF. Justificați răspunsul.



C. 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 <=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)

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)** \mathbf{x} =g => T **b)** \mathbf{x} =h => NIL