Programare logică și funcțională - examen scris -

<u>Notă</u>

- 1. Subjectele 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(L)

(COND

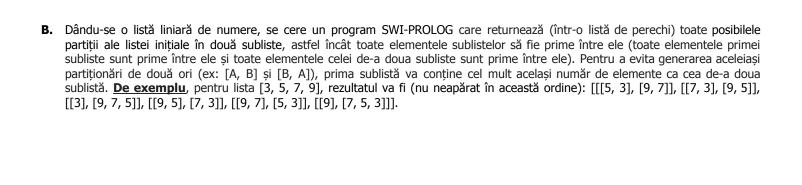
((NULL L) 0)

(> (F (CDR L)) 2) (+ (F (CDR L)) (CAR L)))

(T (+ (F (CDR L)) 1))

)
```

Rescrieți această definiție pentru a evita apelul recursiv repetat **(F (CDR L))**. Nu redefiniți funcția. Nu folosiți SET, SETQ, SETF. Justificați răspunsul.



C. Să se scrie un program PROLOG care generează lista submulţimilor formate cu elemente unei liste listă de numere întregi, având suma elementelor număr impar și număr impar de elemente impare. Se vor scrie modelele matematice și modelele de flux pentru predicatele folosite.

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

D.	Să se substituie valorile numerice cu o valoare e dată, la orice nivel al unei liste neliniare. Se va folosi o funcție MAP. Exemplu , pentru lista (1 d (2 f (3))), e =0 rezultă lista (0 d (0 f (0))).