

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 LISP

```
(DEFUN F(L)
  (COND
    ((NULL L) 0)
    ((> (F (CAR L)) 1) (F (CDR L)))
    (T (+ (F (CAR L)) (F (CDR L)))))
  )
)
```

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

- B. Dându-se o listă neliniară formată din numere mai mari sau egale cu 2, se cere un program SWI-PROLOG care să înlocuiască fiecare număr neprim cu suma divizorilor săi proprii. Repetați procesul până când lista rămâne doar cu numere prime. **De exemplu.** pentru lista [10, 20, 30, 40] rezultatul va fi [7, 7, 41, 7] (lista inițială devine la început [7, 21, 41, 49], apoi [7, 10, 41, 7] iar final [7, 7, 41, 7]). Va trebui să returnați doar lista finală.

- C. Dându-se o listă formată din numere întregi, să se genereze lista submulțimilor cu **k** elemente în progresie aritmetică. Se vor scrie modelele matematice și modelele de flux pentru predicatele folosite.

Exemplu pentru lista $L=[1,5,2,9,3]$ și $k=3 \Rightarrow [[1,2,3],[1,5,9],[1,3,5]]$ (nu neapărat în această ordine)

- D. Un arbore n-ar se reprezintă în LISP astfel (nod subarbore1 subarbore2). Se cere să se determine lista nodurilor 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 => (g d) **b)** k=5 => ()