

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 calculeze diferența dintre cel mai mic maxim din subliste și cel mai mare dintre valorile minime din subliste. Se presupune că lista de intrare conține cel puțin o sublistă. **De exemplu**, pentru lista `[[4, 2, 18], 7, 2, -3, [6, 9, 11, 3], 4, [5, 9, 19]]` rezultatul va fi 6.

- C. Dându-se o listă formată din numere întregi, să se genereze în PROLOG lista aranjamentelor cu **N** elemente care se termină cu o valoare impară și au suma **S** dată. Se vor scrie modelele matematice și modelele de flux pentru predicatele folosite.

Exemplu- pentru lista $L=[2,7,4,5,3]$, $N=2$ și $S=7 \Rightarrow [[2,5], [4,3]]$ (nu neapărat în această ordine)

D. Un arbore n-ar se reprezintă în LISP astfel (nod subarbore1 subarbore2)

Se cere să se înlocuiască nodurile de pe nivelul **k** din arbore cu o valoare **e** dată. Nivelul rădăcinii se consideră a fi 0.

Se va folosi o funcție MAP.

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

- a) $k=2 \Rightarrow (a (b (h)) (c (h (e)) (h)))$
- b) $k=4 \Rightarrow (a (b (g)) (c (d (e)) (f)))$