

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)) 2) (+ (CAR L) (F (CDR L))))
    (T (F (CAR 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 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. Pentru o valoare **N** dată, să se genereze lista permutărilor cu elementele $N, N+1, \dots, 2*N-1$ având proprietatea că valoarea absolută a diferenței dintre două valori consecutive din permutare este ≤ 2 . Se vor scrie modelele matematice și modelele de flux pentru predicatele folosite.

D. Se dă o listă neliniară și se cere înlocuirea valorilor numerice pare cu numărul natural succesor. **Se va folosi o funcție MAP.**

Exemplu pentru lista (1 s 4 (2 f (7))) va rezulta (1 s 5 (3 f (7))).