

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 predicat PROLOG **f(integer, integer)**, având modelul de flux (i, o):

$f(20, -1):-!$.

$f(I,Y):-J \text{ is } I+1, \underline{f(J,V)}, V>0, !, K \text{ is } J, Y \text{ is } K.$

$f(I,Y):-J \text{ is } I+1, \underline{f(J,V)}, Y \text{ is } V-1.$

Rescrieți această definiție pentru a evita apelul recursiv **f(J,V)** în ambele clauze. Nu redefiniți predicatul. Justificați răspunsul.

- B.** Dându-se o listă neliniară care conține atomi numerici și nenumeriți, se cere un program Lisp care concatenează fiecare (sub)listă liniară în care numărul atomilor numerici este egal cu numărul atomilor nenumeriți cu lista în care se află. Se repetă acest proces până lista nu mai poate fi modificată. De exemplu, pentru lista (A B (4 A 3) 11 (5 (A (B 3) (C 10) 1) (4 A) X Y Z)5) rezultatul va fi (A B (4 A 3) 11 (5 A B 3 C 10 1 4 A X Y Z) 5).

- C. Să se scrie un program PROLOG care generează lista submulțimilor cu **N** elemente, cu elementele unei liste, astfel încât suma elementelor dintr-o submulțime să fie număr par. Se vor scrie modelele matematice și modelele de flux pentru predicatele folosite.

Exemplu- pentru lista $L=[1, 3, 4, 2]$ și $N=2 \Rightarrow [[1,3], [2,4]]$

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