**I Laborator 4 PLF L1**

**II** **Programare recursivă în Lisp (1)**

**III Predat la:** 25.11.2021

**IV Problema 1**

a) Să se insereze într-o listă liniară un atom *a* dat după al 2-lea, al 4-lea, al 6-lea, … element.

b) Definiți o funcție care obține dintr-o listă dată lista tuturor atomilor care apar, pe orice nivel, dar în ordine inversă. De exemplu: (((A B) C) (D E)) => (E D C B A).

c) Definiți o funcție care întoarce cel mai mare divizor comun al numerelor dintr-o listă neliniară.

d) Să se scrie o funcție care determină numarul de apariții ale unui atom dat intr-o listă neliniară.

**V Model matematic recursiv**

inserare([l1, l2, …, ln], a, i)=

invers([a1,a2, ..., an])=

cmmdc(a b) =

cmmdc\_list([a1,a2, ..., an]) =

aparitii([l1, l2, …, ln], a)=

**VI Cod sursă**

(defun inserare(l a i)

(cond

((AND (= (mod i 2) 1) (null l)) (LIST a))

((null l) nil)

((= (mod i 2) 0) (APPEND (LIST (CAR l)) (LIST a) (inserare (CDR l) a (+ i 1))))

(t (APPEND (LIST (CAR l)) (inserare (CDR l) a (+ i 1))))

)

)

(defun invers(l)

(cond

((null l) nil)

((ATOM (CAR l)) (APPEND (invers (CDR l)) (LIST (CAR l))))

((LISTP (CAR l)) (APPEND (invers (CDR l)) (invers (CAR l))))

)

)

**c)**

(defun cmmdc(a b)

(cond

((= b 0) a)

(t (cmmdc b (mod a b)))

)

)

(defun cmmdc\_list(l)

(cond

((null l) 0)

((LISTP (CAR l)) (cmmdc (cmmdc\_list (CAR l)) (cmmdc\_list (CDR l))))

(t (cmmdc (CAR l) (cmmdc\_list (CDR l))))

)

)

**d)**

(defun aparitii(l a)

(cond

((null l) 0)

((LISTP (CAR l)) (+ (aparitii (CAR l) a) (aparitii (CDR l) a)))

((equal (CAR l) a) (+ 1 (aparitii (CDR l) a)))

(t (aparitii (CDR l) a))

)

)

**VII Exemple de testare**

Cazurile sunt scrise de forma(date de intrare=>rezultat).

1. *inserare(l a i)*

1. Când l = 0 => (inserare `() 10 1) = [10]

2. (inserare `(1 2 3 4 5) 10 1) = [1 2 10 3 4 10 5]

(inserare `(1 2 3) 10 1) = [1 2 10 3]

(inserare `(1 2 3 4 5 6) 10 1) = [1 2 10 3 4 10 5 6 10]

1. *invers(l)*

1. Când l = 0 => (invers `()) = NIL

2. (invers `(1 2 3 4) = [4 3 2 1]

(invers `(1 (2 3) (4)) = [4 3 2 1]

1. *cmmdc(a b) si cmmdc\_list(l)*

1.1. Când b = 0 => cmmdc( a 0) = a

2.1. (cmmdc 12 18) = 6

1.2. Când l = 0 => (cmmdc\_list `()) = 0

2.1. (cmmdc\_list `(8 (16) 32)) = 8

(cmmdc\_list `((4 8) (16) 32)) = 4

1. *aparitii(l a)*

1. Când l = 0 => (aparitii `() 1) = 0

2. (aparitii `(1 1 5 6 1 9 1 2 2) 1) = 4

(aparitii `(1 1 5 6 1 9 1 2 2) 2) = 2