Problema 9

1. Sa se scrie o functie care intoarce diferenta a doua multimi.

apare(e, l1,…ln) = {[], daca l e vida

{true, daca l1 = e

{diferenta(e, l2,…ln), altfel

diferenta(l1,…,ln, k1,…,kn) = {[], daca prima lista e vida

{diferenta(l2,…,ln, k1,…km) daca apare(l1,k1,…,km) = true

{l1 (+) diferenta(l2,…,ln,k1,…km) altfel.

1. Definiti o functie care inverseaza o lista impreuna cu toate sublistele sale de pe orice nivel.

invers(l1,l2,…,ln) = {[], daca l e vida

{invers(l2,…,ln) (+) l1, daca l1 e atom

{invers(l2,…,ln) (+) invers(l1), altfel

c) Dandu-se o lista, sa se construiasca lista primelor elemente ale tuturor elementelor lista ce au un numar impar de elemente la nivel superficial. Exemplu: (1 2 (3 (4 5) (6 7)) 8 (9 10 11)) => (1 3 9).

numara(l1,…,ln) = {() daca lista e vida

{1 + numara(l2,…,ln) altfel

Primele\_aux(l1,…,ln) = {() daca lista e vida

{l1[0] (+) primele\_aux(l2,…,ln), daca l1 e lista si numara(l1) e impar

{primele\_aux(l2,…,ln) altfel

Primele(l1,…,ln) = {l1 (+) primele\_aux(l1,…,ln), daca count(l1,…,ln) e impar

{primele\_aux(l1,…,ln) altfel

d) Sa se construiasca o functie care intoarce suma atomilor numerici dintr-o lista, de la nivelul superficial.

Suma(l1,…,ln) = {0 daca lista e vida

{l1 + suma(l2,…,ln), daca l1 e numar

{suma(l2,…,ln) altfel

Problema 1

1. Se da un arbore de tipul (1). Sa se afiseze calea de la radacina pana la un nod x dat.

Parcurg\_st(l1,…,ln, nv, nm) = {() daca lista e vida

{() daca nv = 1 + nm

{l1 (+) l2 (+) parcurg\_st(l3,…,ln, nv + 1, nm + l2) altfel

Stang(l1,…,ln) = parcurg\_st(l3,…ln, 0, 0)

Parcurg\_dr(l1,…,ln, nv, nm) = {() daca lista e vida

{l1,…,ln, daca nv = 1 + nm

{l1 (+) l2 (+) parcurg\_dr(l3,…,ln, nv + 1, nm + l2) altfel

Drept(l1,…,ln) = parcurg\_dr(l3,…ln, 0, 0)

Parcurgere\_aux(l1,…,ln, nod, rez) = {() daca lista e vida

{rez (+) nod, daca l1 = nod

{parcurgere\_aux(stang(l1,…,ln), nod, rez (+) l1) (+)

parcurgere\_aux(drept(l1,…,ln), nod, rez (+) l1) altfel

parcurgere(l1,…,ln, nod) = parcurgere\_aux(l1,…,ln, nod, ())