MD4\_2

Dainis Tillers

dt08050

# Programmas teksts

% salīdzināšanas predikāti, tiek izmantoti, jo termini ir salikti no 3 dažādām vērtībām

cmp\_1((\_, X, \_), (\_, Y, \_)) :- X =< Y, !.

cmp\_2((\_, X, \_), (\_, Y, \_)) :- X > Y, !.

% tā kā lapu vērtības tiek padotas kā sadalītas vērtības, tad, lai tās sakārtotu var izmantot merge sort algoritma merge daļu

merge([Xh|Xt], [Yh| Yt], [Xh|Z]) :- cmp\_1(Xh, Yh), !,merge(Xt,[Yh|Yt], Z).

merge([Xh|Xt], [Yh| Yt], [Yh|Z]) :- cmp\_2(Xh, Yh), !,merge([Xh|Xt], Yt, Z).

merge(X, [], Z) :- Z = X.

merge([], Y, Z) :- Z = Y.

% atrod minimālo vērtību sarakstā, tukšam sarakstam tā ir 0

min\_v([], X) :- X is 0.

min\_v(A, X) :- min\_list(A, X).

% atrod maksimālo vērtību sarakstā, tukšam sarakstam tā ir 0

max\_v([], X) :- X is 0.

max\_v(A, X) :- max\_list(A, X).

% koka apstaigāšanas funkcijas, ja virsotne ir lapa vai arī ar vienu bērnu tad rezultātam nonākot tajā, tas ir jau sakārtots

aa(n0(A), X) :- min\_v(A, Min), max\_v(A, Max), length(A, Len), X = [(Len, Min, Max)]. % atrod minālo, maksimālo vērtību un saraksta garumumu

aa(n1(T1), X) :- aa(T1, Y), append(Y, [], X).

aa(n2(T1, T2), X) :- aa(T1, Y), aa(T2, Z), merge(Y, Z, X). % tā kā šajā gadījumā ir divi rezultāti tad tie ir jāsavieno kopā, lai rezultāts būtu sakārtots

% veic katra saraksta elementa kāpināšanu kvadrātā, darbs notiek rekursīvi

double\_list([], X) :- X = [].

double\_list([Xh | Xt], X) :- Y is Xh \* Xh, double\_list(Xt, Z), append([Y], Z, X).

% veic koka apstaigāšanu un rekonstruēšanu

bb(n0(A), X) :- double\_list(A, Y), X = n0(Y).

bb(n1(T1), X) :- bb(T1, Y), X = n1(Y).

bb(n2(T1, T2), X) :- bb(T1, Y), bb(T2, Z), X = n2(Y, Z).

# Testpiemēri

1. aa(n1(n1(n2(n1(n0([])), n2(n1(n0([1, 2, 3])), n0([2]))))) , X).

X = [ (0, 0, 0), (3, 1, 3), (1, 2, 2)] .

bb(n1(n1(n2(n1(n0([])), n2(n1(n0([1, 2, 3])), n0([2]))))) , X).

X = n1(n1(n2(n1(n0([])), n2(n1(n0([1, 4, 9])), n0([4]))))).

1. aa(n2(n2(n2(n0([3]), n0([4])), n0([1])), n0([2])), X).

X = [ (1, 1, 1), (1, 2, 2), (1, 3, 3), (1, 4, 4)].

bb(n2(n2(n2(n0([3]), n0([4])), n0([1])), n0([2])), X).

X = n2(n2(n2(n0([9]), n0([16])), n0([1])), n0([4])).

1. aa(n2(n2(n1(n0([2,3,4,0])),n0([2,-1,3])),n0([])),X).

X = [ (3, -1, 3), (4, 0, 4), (0, 0, 0)]

bb(n2(n2(n1(n0([2,3,4,0])),n0([2,-1,3])),n0([])),X).

X = n2(n2(n1(n0([4, 9, 16, 0])), n0([4, 1, 9])), n0([])).