

karai - torugai:

eljárás karai (D_0 : egész; r_1, r_2, r_3 : karakter);

ha $D_0 > \emptyset$ akkor

karai ($D_0 - 1, r_1, r_2, r_3$)

ki: D_0 , "korug átírása", r_1 , " \rightarrow ", r_3

karai ($D_0 - 1, r_2, r_3, r_1$);

k vége;

vége;

Rekurzív bináris keresés:

függvény rekürzés (A : tömbtípus; Adat: elemtípus;

hely: egész; e, v : egész); logikai;

változó k : egész;

ha $e > v$ akkor

rekürzés := hamis.

különben

$k := \text{egészrész}((e+v)/2)$;

elgazdas

amikor $\text{Adat} < A[k]$:

rekürzés := rekürzés($A, \text{Adat}, \text{hely}, e, k-1$);

amikor $\text{Adat} > A[k]$:

rekürzés := rekürzés($A, \text{Adat}, \text{hely}, k+1, v$)

különben

hely := k ;

rekürzés := igaz;

e vége;

k vége;

vége;

Gyorsrendezés (Quicksort):

eljárás gyorsrend(A : tömbtípus; felső, alsó: egész);

változó i, j : egész;

változó x , csere: elemtípus;

$i :=$ alsó; $j :=$ felső;

$x := A[(\text{felső} + \text{alsó})/2]$;

ismétel

amíg $A[i] < x$ ismétel

$i := i + 1$;

a vége;

amíg $A[j] > x$ ismétel

$j := j - 1$;

a vége;

ha $i < j$ akkor

$\text{csere} := A[i]$; $A[i] := A[j]$; $A[j] := \text{csere}$;

a vége;

ha $i \leq j$ akkor

$i := i + 1$; $j := j - 1$;

a vége;

i vége $i > j$ esetén

ha alsó $< j$ akkor

gyorsrend(A , alsó, j);

a vége;

ha $i < \text{felső}$ akkor

gyorsrend(A , i , felső);

a vége;

vége;

LISTA:

1. Láncolt lista:

deklaráció:

tipus mutatótipus: egész;

H_{tipus} értéktípus: tárolható - értékel - típusa;

H_{tipus} listacéltípus: rekord (

 'érték' : értéktípus;

 'köv' : mutatótipus;

)

konstans maxelem = maximális - elemszám;

változó elem : tömb [1.. maxelem] listacéltípus;

konstans végjel = 0;

változó szabad : mutatótipus;

szabad lista inicializálása:

előadás szabadrezd;

változó i: egész;

szabad := 1;

allos i := 1.. maxelem - 1 ismételt

 elem [i]. köv := i + 1;

c vége;

elem [maxelem]. köv := végjel;

vége;

Elem lefoglalása (törés a szabad listából):

függvény lefoglal (hely: mutatótípus): logikai;

ha szabad <> végjel akkor

hely := szabad;

szabad := elem [szabad].köv;

lefoglal := igaz

különben

lefoglal := hamis;

és vége;

vége;

Elem felszabadítása (vissafűzés a szabad listába):

eljárás (hely: mutatótípus);

ha hely <> végjel akkor

elem [hely].köv := szabad;

szabad := hely;

és vége;

vége;

lista inicializálása:

eljárás listarész (fej: mutatótípus);

fej := végjel;

vége;

lista feldolgozása:

eljárás listafeldolgoz (fej: mutatótípus);

változó p: mutatótípus;

p := fej;

amíg p <> végjel ismételd

elem [p]. érték feldolgozása

p := elem [p]. követő;

a vége;

vége;

Keresés a listában:

függvény listakeres (fej: mutatótípus; Adat: értéktípus;
 hely: mutatótípus): logikai;

változó p: mutatótípus;

p := fej;

amíg (p <> végjel) és (elem [p]. érték <> Adat) ismételd

p := elem [p]. követő;

a vége;

hely := p;

listakeres := (p <> végjel);

vége;

Mozaikfüzés a lista elején:

függvény listafüzelő (fej : mutatótípus; Adat: értékípus): logikai;

változó p : mutatótípus;

ha lefoglal (p) akkor

elem [p]. érték := Adat;

elem [p]. elő := fej ;

$fej := p$;

listafüzelő := igaz

különben

listafüzelő := hamis,

a vége;

vége;

Törés a lista elején:

függvény listatörőelő (fej : mutatótípus): logikai;

változó p : mutatótípus;

ha $fej \wedge$ vége akkor

$p := fej$;

$fej :=$ elem [fej]. elő;

felszabadít (p);

listatörőelő := igaz.

különben

listatörőelő := hamis;

a vége;

vége;

Középső adott elem mögé:

függvény listafüzmögé (emögé: mutatótípus; Adat: értéktípus):
logikai;

változó p: mutatótípus;

ha (emögé <> végjel) és előglal (p) akkor

elem [p].érték := Adat;

elem [p].elő := elem [emögé].elő;

elem [emögé].elő := p;

listafüzmögé := igaz

különben

listafüzmögé := hamis;

h vége;

vége;

Törés adott elem mögül:

függvény listatörésmögül (emögül: mutatótípus): logikai;

változó p: mutatótípus;

ha (emögül <> végjel) és (elem [emögül].elő <>
végjel) akkor

p := elem [emögül].elő;

elem [emögül].elő := elem [p].elő;

felsabadít (p);

listatörésmögül := igaz

különben

listatörésmögül := hamis;

h vége;

vége;

Középfűzés a lista végén:

függvény listafűzés (fej: mutatótípus; Adat: értéktípus): logikai;

változó p: mutatótípus;

ha fej = végjel akkor

listafűzés := listafűző (fej, Adat)

különben

p := fej;

amíg elem [p]. előv <> végjel ismétli

p := elem [p]. előv;

a vége;

listafűzés := listafűző (p, Adat);

h vége;

vége;