

Algoritmusok és adatszerkezetek zárthelyi megoldások

2004. május 18.

Lőrentey Károly

1. feladat: (12 pont)

Szemléltessük a *kupacrendezés* működését az alábbi kiinduló adatokkal:

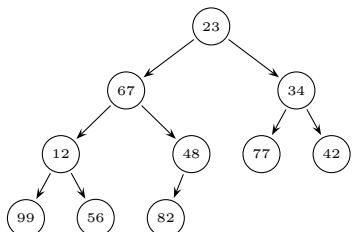
23, 67, 34, 12, 48, 77, 42, 99, 56, 82

Adjuk meg a tömb állapotát a kupacra hozás után, és rajzoljuk fel a rendező algoritmus egyes lépéseinek eredményeképpen előálló állapotokat.

Megoldás: A feladat a *tömbállapotok* megadása volt. A szemléletesség kedvéért a tömbök után megadjuk a tömb által reprezentált bináris fát is. A kiinduló állapot:

0:

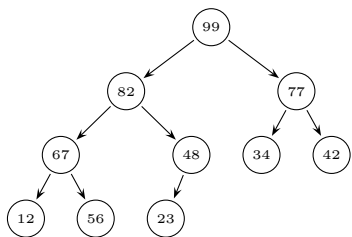
23	67	34	12	48	77	42	99	56	82
----	----	----	----	----	----	----	----	----	----



Kupaccá alakítás után:

1:

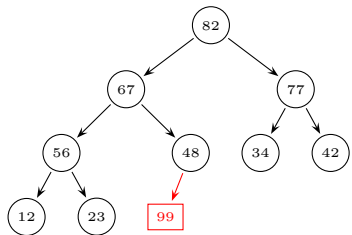
99	82	77	67	48	34	42	12	56	23
----	----	----	----	----	----	----	----	----	----



A rendezés lépései (a süllyesztés után):

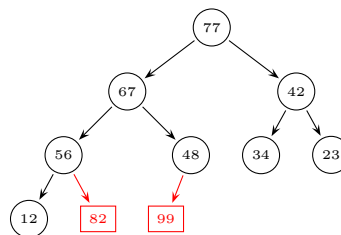
2:

82	67	77	56	48	34	42	12	23	99
----	----	----	----	----	----	----	----	----	----



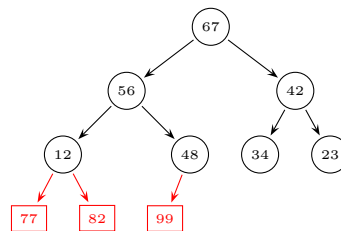
3:

77	67	42	56	48	34	23	12	82	99
----	----	----	----	----	----	----	----	----	----



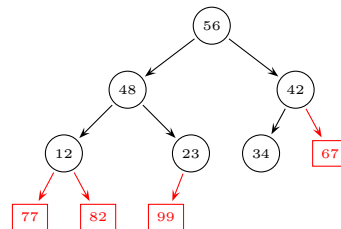
4:

67	56	42	12	48	34	23	77	82	99
----	----	----	----	----	----	----	----	----	----



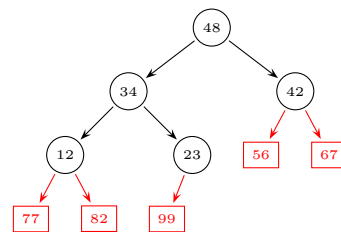
5:

56	48	42	12	23	34	67	77	82	99
----	----	----	----	----	----	----	----	----	----



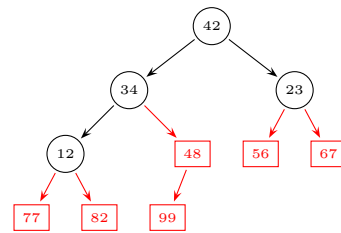
6:

48	34	42	12	23	56	67	77	82	99
----	----	----	----	----	----	----	----	----	----



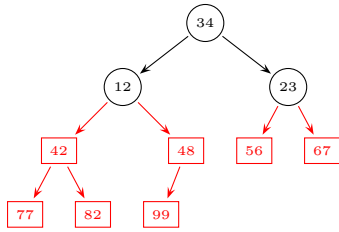
7:

42	34	23	12	48	56	67	77	82	99
----	----	----	----	----	----	----	----	----	----



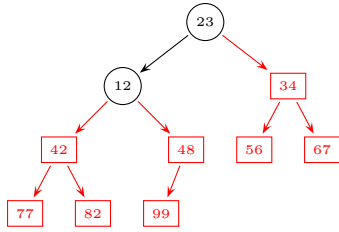
8:

34	12	23	42	48	56	67	77	82	99
----	----	----	----	----	----	----	----	----	----



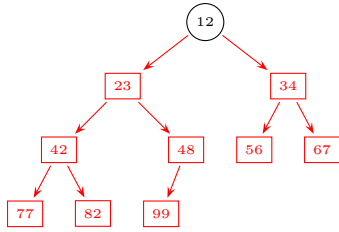
9:

23	12	34	42	48	56	67	77	82	99
----	----	----	----	----	----	----	----	----	----



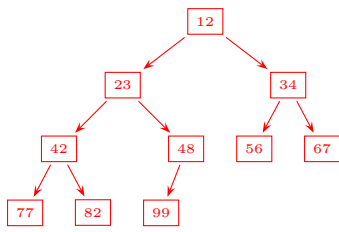
10:

12	23	34	42	48	56	67	77	82	99
----	----	----	----	----	----	----	----	----	----



11:

12	23	34	42	48	56	67	77	82	99
----	----	----	----	----	----	----	----	----	----



2. feladat: (12 pont)
Demonstráljuk a *számláló rendezés* működését az alábbi kiinduló adatokkal!

3, 1, 2, 2, 4, 2, 4, 4, 1, 1, 2, 4, 5, 2, 3, 1, 1, 2, 2, 3

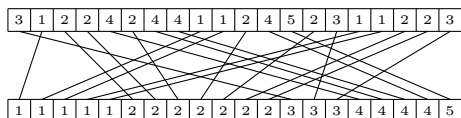
Megoldás: A segédtömb állapota a rendezendő adatokat tartalmazó tömb első felolvasása után:

5	7	3	4	1
---	---	---	---	---

A segédtömb állapota az összegzés után:

5	12	15	19	20
---	----	----	----	----

A céltömb és a forrástömb elemeinek megfeleltetése:



3. feladat: (12 pont)

Legyen a H hasítótábla mérete $m = 10$ és alkalmazunk *szorzásos hasítófüggvényt* $A = 0.618$ állandóval. Tegyük fel, hogy nyílt címzést használunk *lineáris ütközésfeloldással*. Üres táblából kiindulva helyezük el a táblában rendre az alábbi értékeket:

5, 9, 13, 3, 15, 10, 7, 1

Adjuk meg az így kapott, nyolc elemet tartalmazó hasítótáblát. Adjuk meg, hogy a nyolc elem közül az egyik kikereséséhez átlagosan hány táblázathivatkozást kell elvégezni. Ezután töröljük a táblából rendre a 15, 1, 10 értékeket, majd adjuk meg a végeredményül kapott táblázatot.

Segítségül íme az A konstans első néhány többszöröse:

$k - Ak$	$k - Ak$	$k - Ak$
1 — 0.618	6 — 3.708	11 — 6.798
2 — 1.236	7 — 4.326	12 — 7.416
3 — 1.854	8 — 4.944	13 — 8.034
4 — 2.472	9 — 5.562	14 — 8.652
5 — 3.09	10 — 6.18	15 — 9.27

Megoldás: A hasítófüggvény definíciója $h(k) = \lfloor m(kA \bmod 1) \rfloor$, értékei a megadott kulcsokra:

$k - h(k)$	$k - h(k)$
5 — 0	15 — 2
9 — 5	10 — 1
13 — 0	7 — 3
3 — 8	1 — 6

A fenti értékek alapján a táblát kitöltve az alábbi eredményt kapjuk:

0	1	2	3	4	5	6	7	8	9
5	13	15	10	7	9	1		3	

A 13, 10, és 7 értékek beszúrása közben kulcsütközést tapasztaltunk. A 13 és a 7 esetében egyszeri, a 10 esetén másodszori próbálkozásra sikerült üres helyet találnunk a táblában. Ennek alapján a kereséses átlagos táblahivatkozás-igénye $(5 \cdot 1 + 2 \cdot 2 + 1 \cdot 3) / 8 = 12 / 8 = 1.5$.

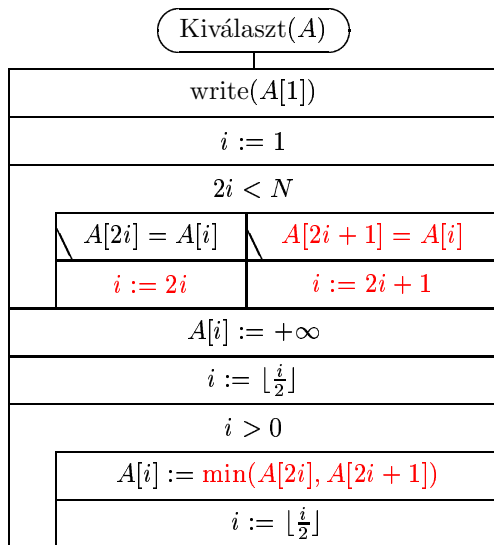
A törlések során egyszerűen csak arra kell ügyelni, hogy a kitörölt elemeket valamilyen módon megkülönböztessük az érintetlen üres celláktól:

0	1	2	3	4	5	6	7	8	9
5	13	×	×	7	9	×		3	

4. feladat: (12 pont)

A mellékelt struktogram a *versenyrendezés* implementációjának része. A megvalósítás a versenyrendezés „bajnokság-fáját” a kupachoz hasonlóan kezelhető A tömbben ábrázolja. A tömb utolsó n eleme tartalmazza a bemenő adatokat. Feltesszük, hogy $n = 2^k$ valamely k egész értékre. A tömb első $2^k - 1$ eleme tartalmazza a „mérkőzések” eredményeit. A tömb méretét az N változóban tároljuk: $N = 2^{k+1} - 1$.

A megadott program a versenyrendezés egy lépését implementálja: a fa gyökerében található értéket kiírja, eltávolítja a fából, majd az ily módon törölt mérkőzéseket újrajátsza. Sajnos a program néhány utasítása elveszett; pótoljuk ezeket!



5. feladat: (12 pont)

A fenti Kiválaszt segéd eljárás felhasználásával adjuk meg a versenyrendezés teljes programját!

Megoldás: Feltehetjük, hogy az A tömb a megfelelő méretben rendezésre áll, és az utolsó n eleme a rendezendő elemeket tartalmazza. A rendező eljárásunk a tömbben visszafelé haladva előállítja a „bajnokság” kezdeti állapotát, majd a Kiválaszt eljárást addig ismétli, amíg az összes elemet ki nem veszi a tömbből.

