

14. tétel

Adatszerkezetek és algoritmusok vizsga

Frissült: 2013. január 28.

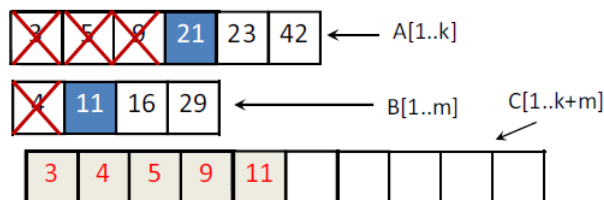
Összefuttatásos rendezés

Alapja két rendezett sorozat összefuttatása.

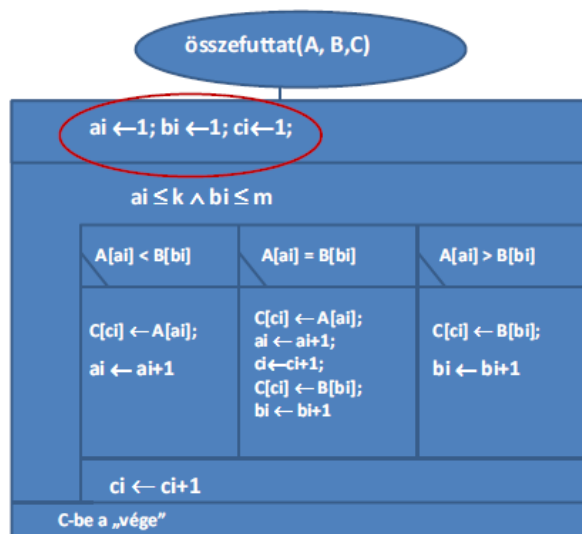
Összefuttatás

Legyen $A[1..k]$ és $B[1..m]$ két rendezett vektor. Továbbá még egy vektor, ahova kerülnek az "összefésült" elemek.

Mindkét vektornak van egy "mutatója". Mindkettőt indítjuk az elejétől. Megnézzük, vajon hol kisebb az érték. Ahol kisebb, azt a kisebb értéket betesszük egy harmadik vektorba, majd lépünk a "mutatóval" egyet. Utána meg- nézzük megint, melyik érték a kisebb, és így tovább, ameddig a két "mutatóval" mindkét vektor végéhez nem értünk. Ha a két érték egyenlő, mind a kettőt betesszük, és mindkettővel lépünk egyet.



Struktogram:



Összehasonlítások száma $k + m - 1$

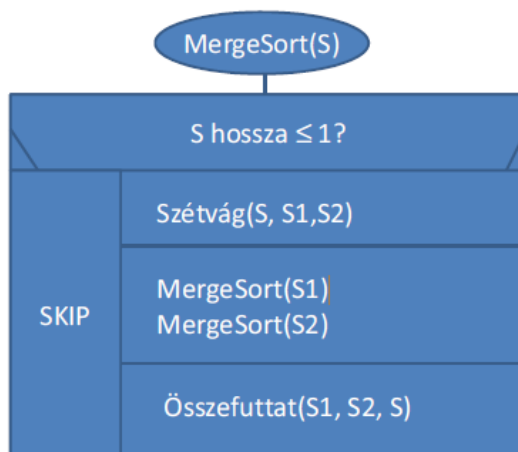
Összefuttatásos rendezés (Mergesort)

Egy $S[1..n]$, $(n > 1)$ tömböt szeretnénk rendezni. Ezeket a lépéseket kell megtenni, rekurzívan:

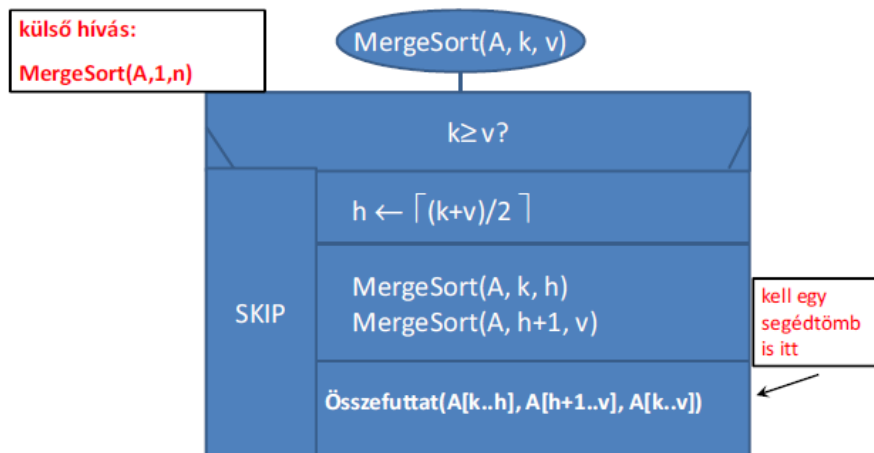
- szétvágjuk két részre (S_1, S_2) – elfelezzük
- ezeket külön-külön rendezzük
- összefuttatjuk

(Az üres és az egyelemű sorozat biztosan rendezett.)

Struktogram:

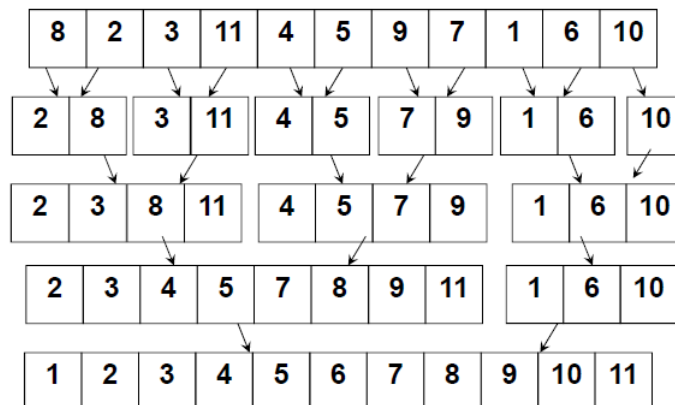


Speciálisan, vektorokra/tömbökre:



Tömbökre szokás egy másik, iteratív stratégiát követni. Minden lépésben végigmegyünk rajta egyre növekvő $h = 1, 2, 2^2, 2^3, \dots, 2^{\lceil \log_2 n \rceil - 1}$ lépésközzel, és az ilyen hosszú szomszédos tömbrészeket fésüljük össze, egy segéd tömböt használva.

Például:



Hatékonyságelemzés

Az összehasonlítások számát becsüljük a *legrosszabb esetben*.

- összefuttat k és m hosszú sorozatot, összefésüléskor $k + m - 1$ összehasonlítást végez:

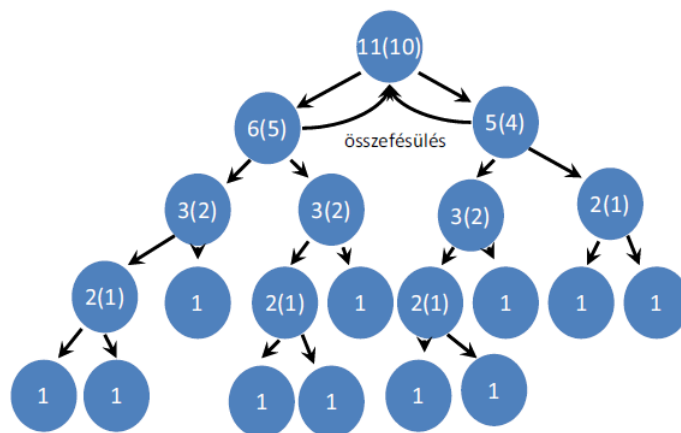
$$M\ddot{O}_{\text{összefuttat}}(k, m) = k + m - 1$$

- ha n a két sorozat együttes hossza:

$$M\ddot{O}_{\text{összefuttat}}(n) = n - 1$$

A *MergeSort* eljárásnál a közel egyenlő szétvágást alapul véve: milyen hosszú részekre vágja az eljárás a (rész) sorozatokat a rekurzív hívások szintjein.

Példa:



$n=11$ -re:

()-ben: öh-k száma

$$MÖ = 10 + (5+4) + (2+2+2+1) + (1+1+1) = 29$$

Az egyenlő szétvágások bináris fája majdnem teljes.

A fa magassága:

$$4 = \lceil \log_2 11 \rceil$$

Általában is:

$$h = \lceil \log_2 n \rceil$$

A fa leveleihez nem tartozik összehasonlítás, így éppen h szinten kell összegezni az összehasonlítások számát. Ezek összege szintenként egyre csökken, mindenütt $\leq n - 1$.

Így

$$MÖMS(n) \leq (n - 1) * \lceil \log_2 n \rceil = \theta(n * \log_2 n)$$

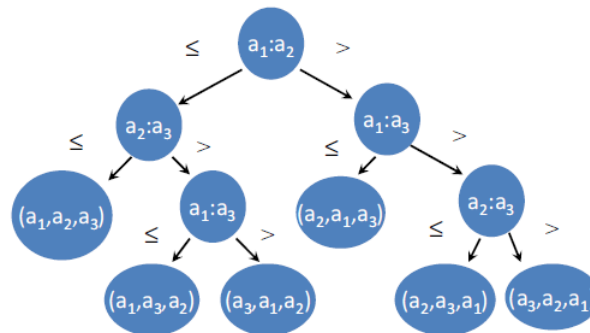
Összehasonlításos rendezőknél mit tudunk mondani a rendezés időigényére? Tudunk-e alsó becslést adni? Ugyanaz, mintha barkóbázva kellene kitalálni, hogy az elemek melyik sorrendje (permutációja) az igazi sorrend. Kezdetben $n!$ lehetséges sorrend jön szóba. Két elemet összehasonlítva, a válasz két részre osztja a sorrendeket.

- Ha pl. azt kapjuk, hogy $x < y$, akkor az olyan sorrendek, amikben x hátrébb van y -nél, már nem jönnek szóba.
- Ha a válasz mindig olyan, hogy minél több sorrend maradjon meg, akkor k kérdés után még szóba jön

$$\frac{n!}{2^k}$$

sorrend.

Döntési fa modell: Az összehasonlítások tekinthetők döntési fáknak, amik a rendezés során történt összehasonlításokat ábrázolják. Tegyük fel, hogy minden elem különböző. Egy belső csúcsot egy $a_i : a_j$ párral jelölhetünk, ahol $1 \leq i, j \leq n$. A levelek egy-egy permutációnak feleltethetők meg.



baloldali részfa: az $a_i \leq a_j$ után szükséges összehasonlítások
 jobboldali részfa: az $a_i > a_j$ után szükséges összehasonlítások
 levél: a rendezett sorrend

Tétel. Bármely n elemet rendező döntési fa magassága:

$$\Omega(n * \log_2 n)$$

A kupacrendezés és az összefésüléses rendezés aszimptotikusan optimális összehasonlító rendezések. Nincs lineáris idejű összehasonlításos rendezés.

Batcher-féle páros-páratlan összefésüléses rendezés

A Mergesort egy változata, szintén egy rekurzív eljárás.

Összefésülés menete

Adottak A és B és U , V tömbök.

1. A **páratlan** és B **páros** indexű elemeit fésüli össze U -ba

$$\begin{array}{l} A = a_1 < a_3 < a_5 \dots < \dots \\ B = b_2 < b_4 < b_6 \dots < \dots \end{array} \quad \Rightarrow \quad U = u_1 < u_2 < u_3 \dots < \dots$$

2. A **páros** és B **páratlan** indexű elemeit fésüli össze V -be

$$\begin{array}{l} A = a_2 < a_4 < a_6 \dots < \dots \\ B = b_1 < b_3 < b_5 \dots < \dots \end{array} \quad \Rightarrow \quad V = v_1 < v_2 < v_3 \dots < \dots$$

Tegyük fel, hogy $k = m$ vagy $k = m+1$

3. Az U és V sorozatokat nem kell összefésülni, hanem elég csak az egymás alatti u_i, v_i párokat 1-1 összehasonlítással a helyes sorrendbe rakni, és így $\{u_1, v_1\}, \{u_2, v_2\}, \{u_3, v_3\}, \dots$ -ből kialakul a helyes sorrend.

Az 1-2 lépés párhuzamosan elvégezhető!

Példa.

A:	1	4	5	7	11	12	14	20
B:	2	3	6	10	13	15	16	17

1. csapat - A páratlan, B páros indexű elemei:

A:	1	5	11	14
B:	3	10	15	17



U:

1	3	5	10	11	14	15	17
---	---	---	----	----	----	----	----

2. csapat - A páros, B páratlan indexű elemei:

A:	4	7	12	20
B:	2	6	13	16

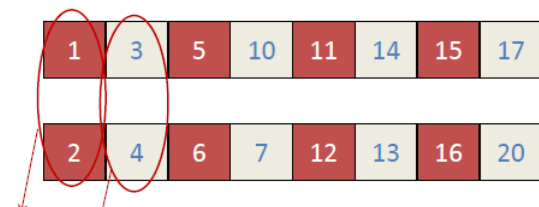


V:

2	4	6	7	12	13	16	20
---	---	---	---	----	----	----	----

$\{u_1, v_1\}, \{u_2, v_2\}, \{u_3, v_3\}$ párosítás:

1	3	5	10	11	14	15	17
2	4	6	7	12	13	16	20



1	2	3	4	5	6	7	1	1	1	1	1	1	1	2
0	1	2	3	4	5	6	7	0	1	2	3	4	5	6

A rekurzió

A 2-2 rövidebb sorozat összefésülését ugyanezzel az eljárással végezzük. Ez rekurzív hívással valósul meg.

Ehhez meg kell adni a legkisebb k és m értéket, amelyre már nem hívja az eljárás önmagát.

Egy 2 hosszú tömböt még szétvágunk két 1 hosszú részre és azokra meghívjuk az összefésülőt, ekkor $k=1$ és $m=1$.

Egy 1 hosszú tömböt azonban már nem fésülünk össze, hanem felismerjük, hogy az már rendezett, így nulla (0) összehasonlítást igényel, ekkor $k=1$, $m=0$.

Helyesség

Állítás. A PP Merge eljárás helyes.

Belátjuk, hogy a leírt eljárás azt a helyes rendezett sorozatot eredményezi, amelyet

$$C = c_1 < c_2 < \dots < c_{m+k}$$

-val jelöltünk

Esetszétválasztással gondoljuk meg:

$$c_1 = \min u_1, v_1, c_2 = \max u_1, v_1$$

Általában, ha $1 \leq i \leq \lfloor (k+m)/2 \rfloor$:

$$c_{2i-1} = \min u_i, v_i, c_{2i} = \max u_i, v_i$$

Vegyük figyelembe a kiegyensúlyozott szétvágást is, azaz azt, hogy $k=m$ vagy $k=m+1$ esetén $\lfloor (k+m)/2 \rfloor = m$.

Ha $k+m$ páratlan, akkor még azt is be kell látni, hogy c_{k+m} is helyesen képződik, hiszen erre akkor a képlet nem vonatkozik.

Batcher sort tömbökre

