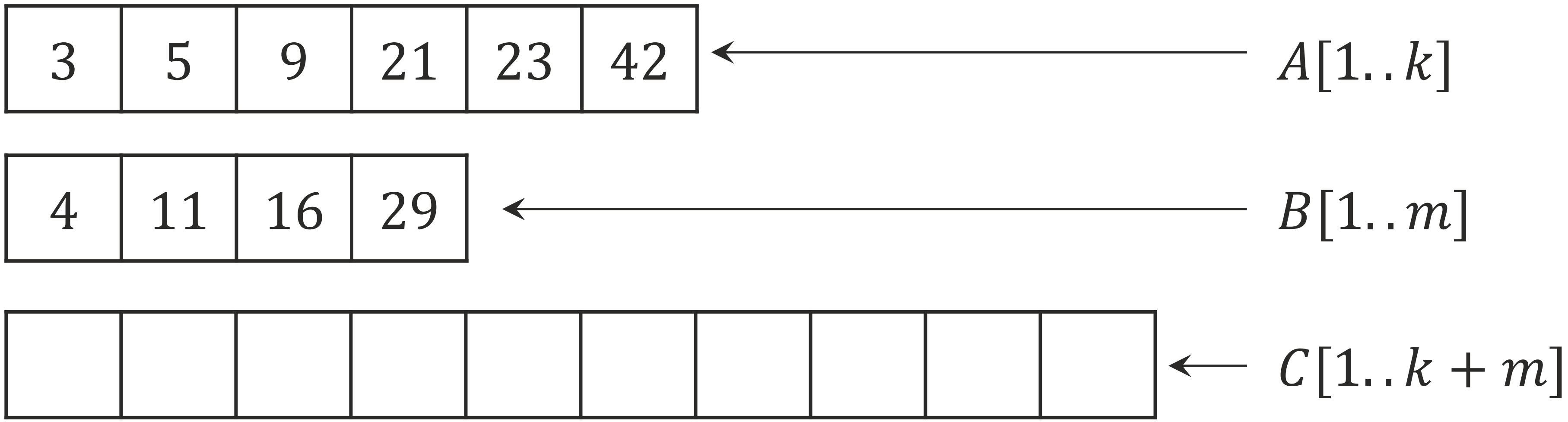
ADATSZERKEZETEK ÉS ALGORITMUSOK

Összefésüléses rendező

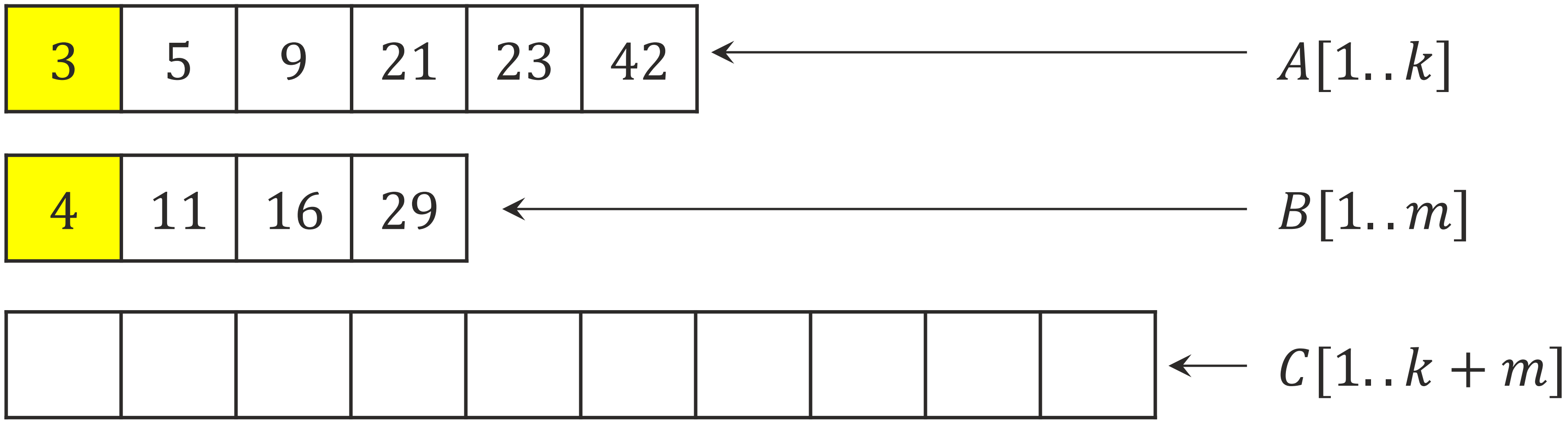
Összefuttatásos rendezés

- Alap: két rendezett sorozat összefuttatása
- Legyen $A[1..k]$ és $B[1..m]$ két rendezett vektor, ekkor:



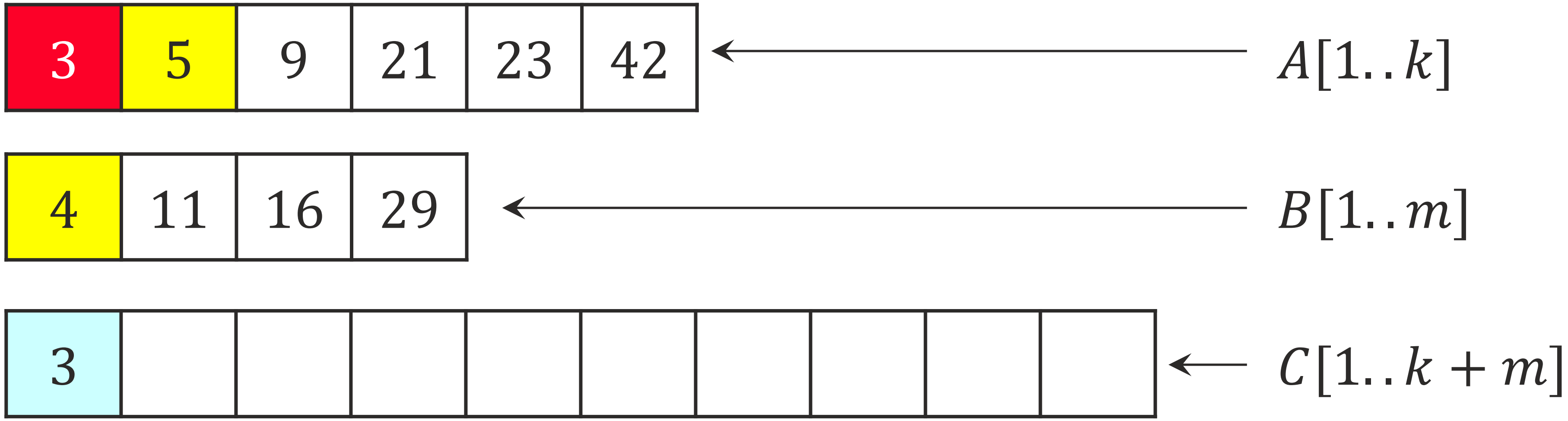
Összefuttatásos rendezés

- Alap: két rendezett sorozat összefuttatása
- Legyen $A[1..k]$ és $B[1..m]$ két rendezett vektor, ekkor:



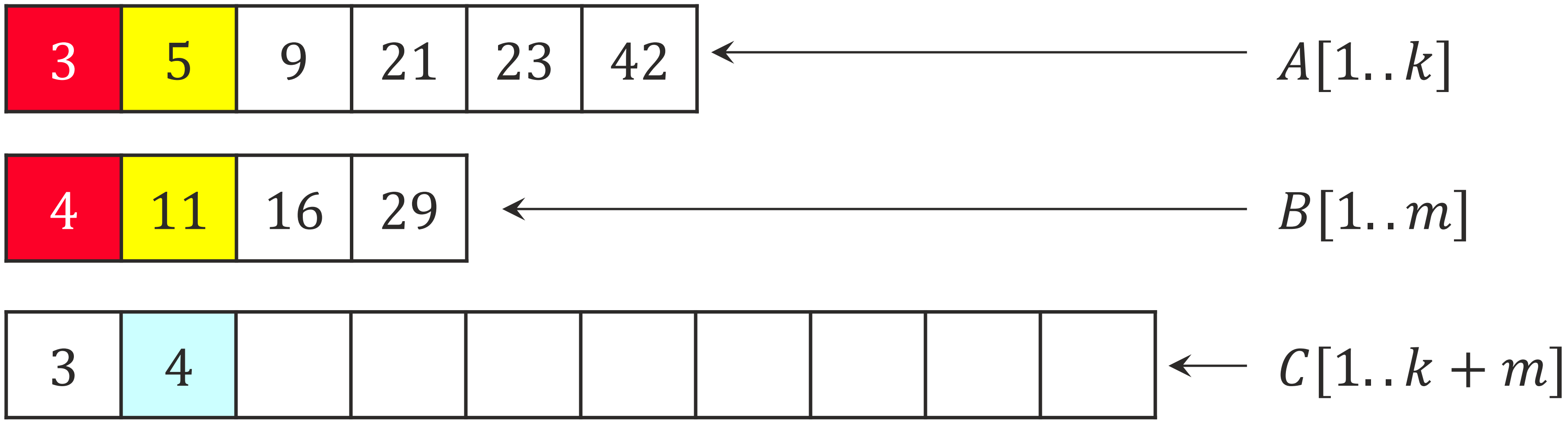
Összefuttatásos rendezés

- Alap: két rendezett sorozat összefuttatása
- Legyen $A[1..k]$ és $B[1..m]$ két rendezett vektor, ekkor:



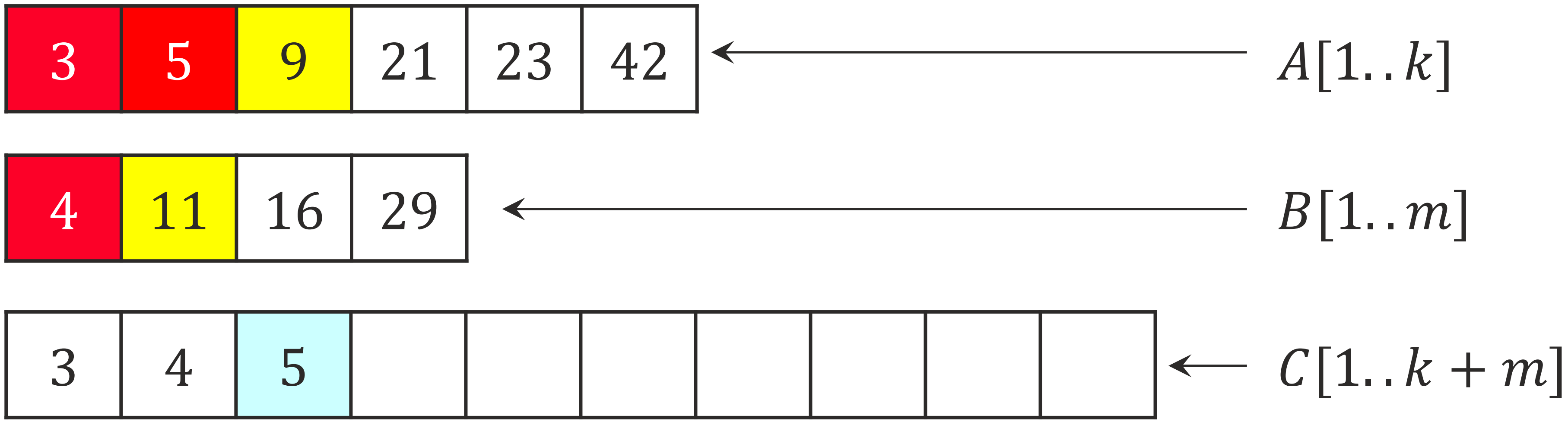
Összefuttatásos rendezés

- Alap: két rendezett sorozat összefuttatása
- Legyen $A[1..k]$ és $B[1..m]$ két rendezett vektor, ekkor:



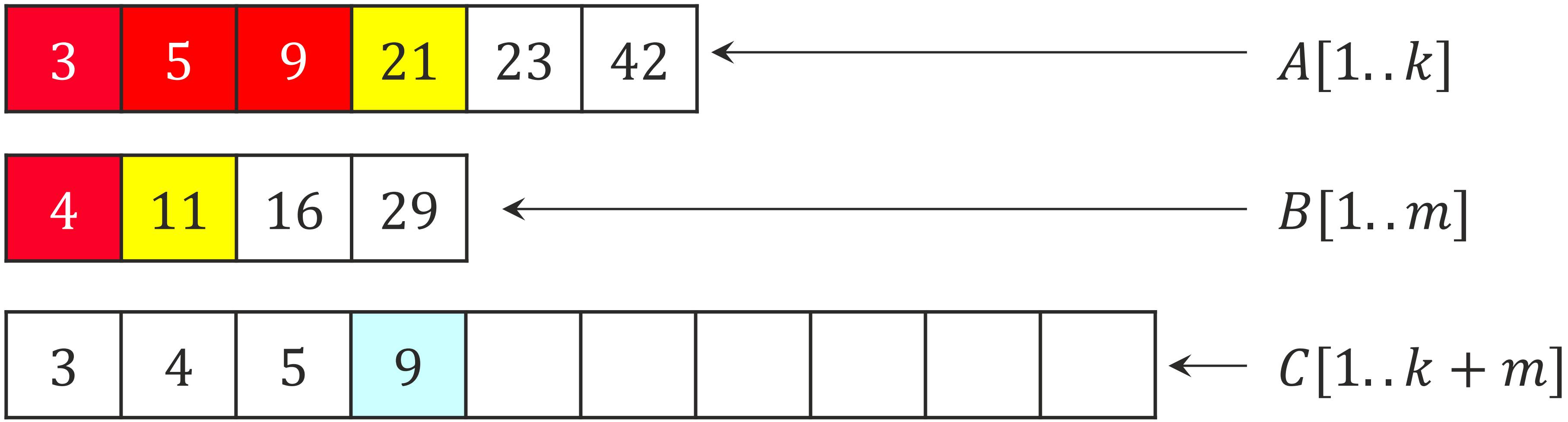
Összefuttatásos rendezés

- Alap: két rendezett sorozat összefuttatása
- Legyen $A[1..k]$ és $B[1..m]$ két rendezett vektor, ekkor:



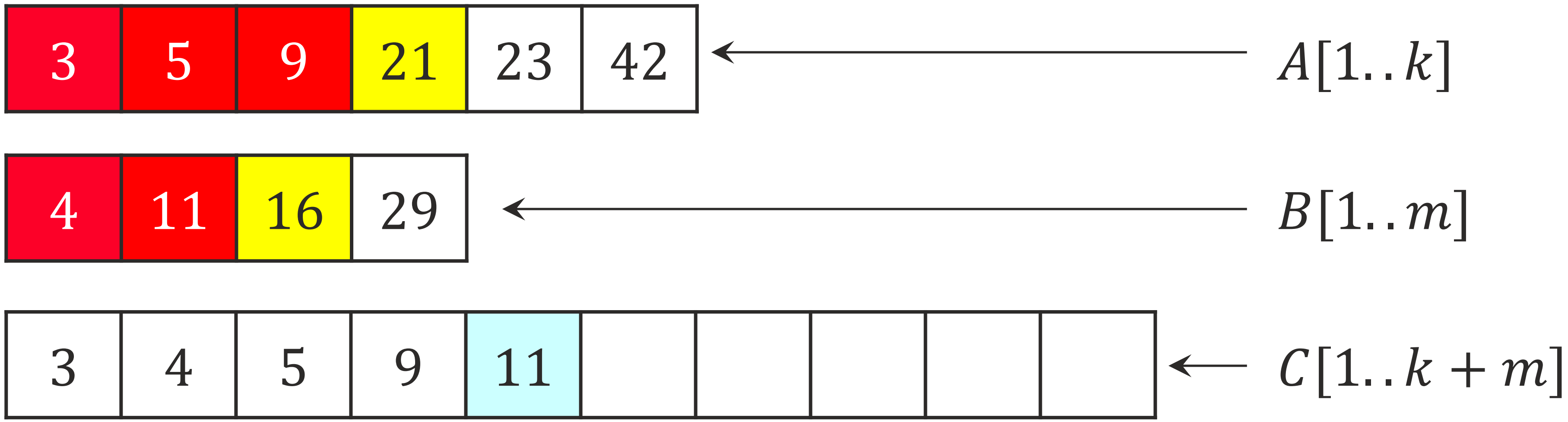
Összefuttatásos rendezés

- Alap: két rendezett sorozat összefuttatása
- Legyen $A[1..k]$ és $B[1..m]$ két rendezett vektor, ekkor:



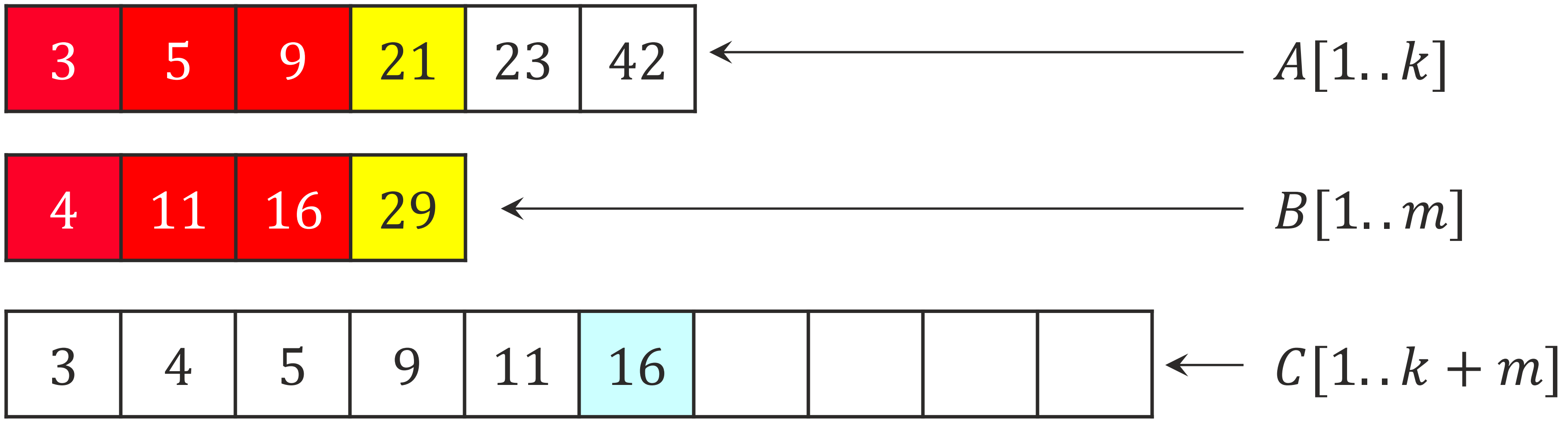
Összefuttatásos rendezés

- Alap: két rendezett sorozat összefuttatása
- Legyen $A[1..k]$ és $B[1..m]$ két rendezett vektor, ekkor:



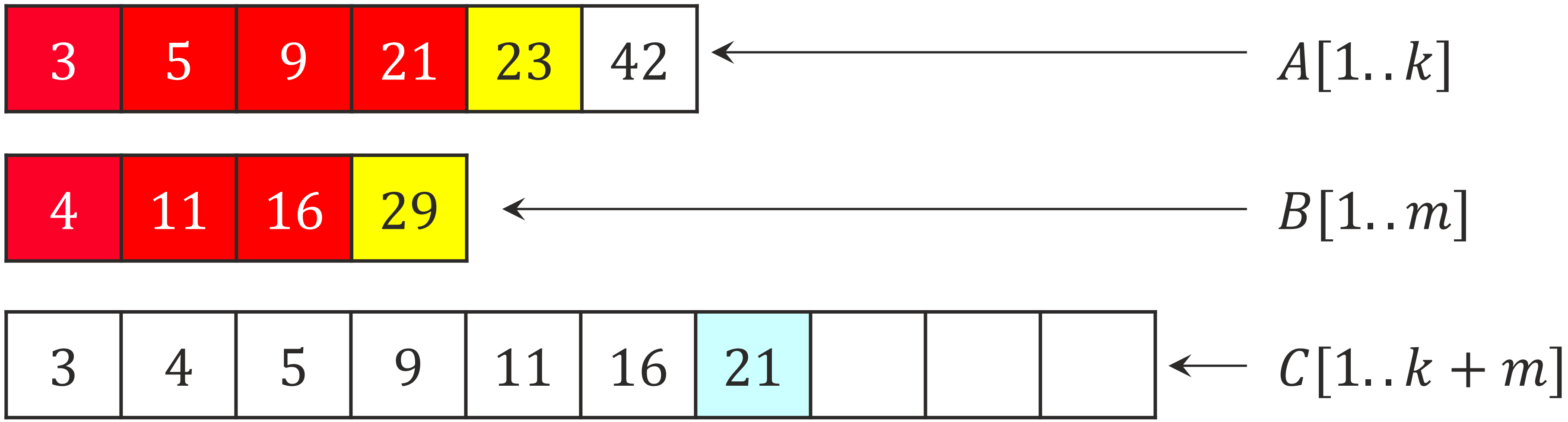
Összefuttatásos rendezés

- Alap: két rendezett sorozat összefuttatása
- Legyen $A[1..k]$ és $B[1..m]$ két rendezett vektor, ekkor:



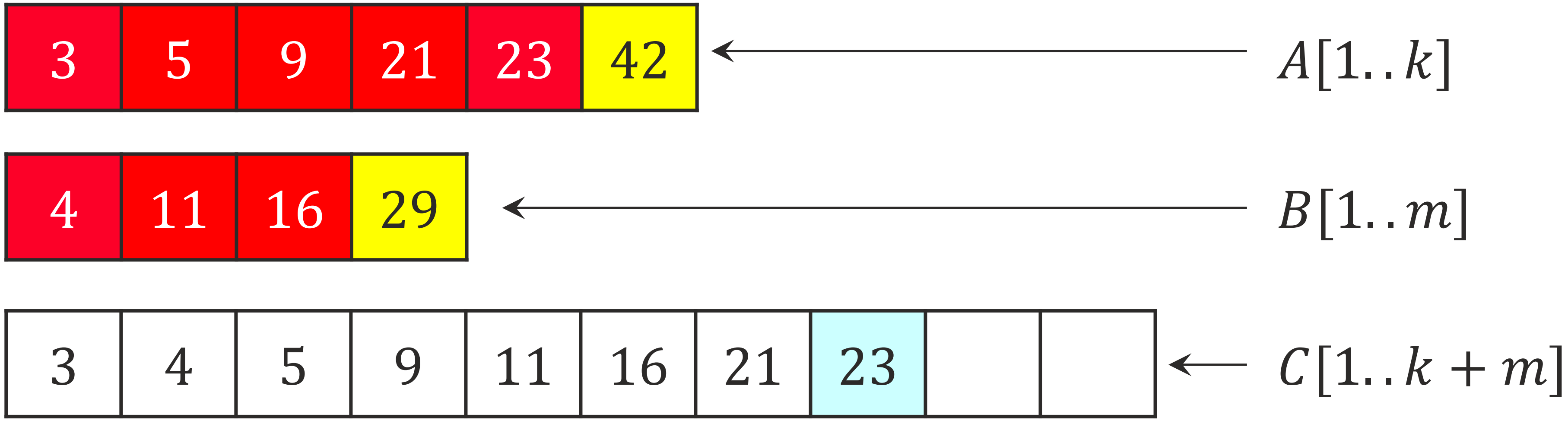
Összefuttatásos rendezés

- Alap: két rendezett sorozat összefuttatása
- Legyen $A[1..k]$ és $B[1..m]$ két rendezett vektor, ekkor:



Összefuttatásos rendezés

- Alap: két rendezett sorozat összefuttatása
- Legyen $A[1..k]$ és $B[1..m]$ két rendezett vektor, ekkor:



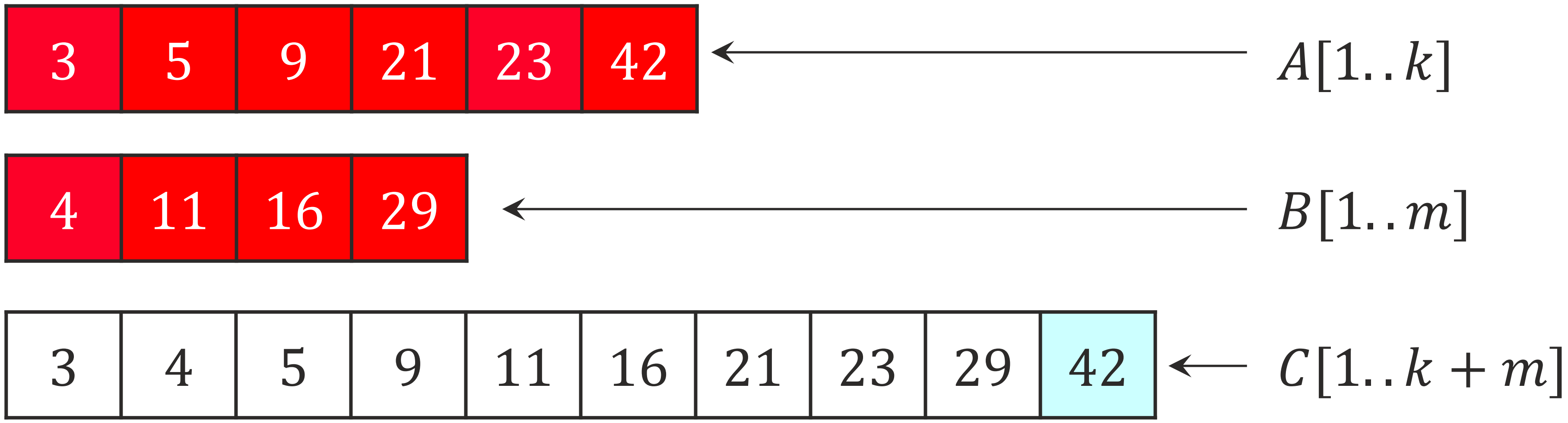
Összefuttatásos rendezés

- Alap: két rendezett sorozat összefuttatása
- Legyen $A[1..k]$ és $B[1..m]$ két rendezett vektor, ekkor:

3	5	9	21	23	42					$\leftarrow A[1..k]$
4	11	16	29							$\leftarrow B[1..m]$
3	4	5	9	11	16	21	23	29		$\leftarrow C[1..k+m]$

Összefuttatásos rendezés

- Alap: két rendezett sorozat összefuttatása
- Legyen $A[1..k]$ és $B[1..m]$ két rendezett vektor, ekkor:



Összefuttatás algoritmusa

- Összefuttat(A, B, C)

$ai \leftarrow 1,\ bi \leftarrow 1,\ ci \leftarrow 1$

$ai \leq k \wedge bi \leq m$

$A[ai] < B[bi]$	$A[ai] = B[bi]$	$A[ai] > B[bi]$
$C[ci] \leftarrow A[ai]$ $ai \leftarrow ai + 1$	$C[ci] \leftarrow A[ai]$ $ai \leftarrow ai + 1$ $ci \leftarrow ci + 1$ $C[ci] \leftarrow B[bi]$ $bi \leftarrow bi + 1$	$C[ci] \leftarrow B[bi]$ $bi \leftarrow bi + 1$

$ci \leftarrow ci + 1$

C tömbbe tegyük a maradékot

Összefuttatás algoritmusa

- Összefuttat(A, B, C)

$ai \leftarrow 1,\ bi \leftarrow 1,\ ci \leftarrow 1$

$ai \leq k \wedge bi \leq m$

- $A[ai] < B[bi]$
  - $C[ci] \leftarrow A[ai]$
  - $ai \leftarrow ai + 1$
- $A[ai] = B[bi]$
  - $C[ci] \leftarrow A[ai]$
  - $ai \leftarrow ai + 1$
  - $ci \leftarrow ci + 1$
  - $C[ci] \leftarrow B[bi]$
  - $bi \leftarrow bi + 1$
- $A[ai] > B[bi]$
  - $C[ci] \leftarrow B[bi]$
  - $bi \leftarrow bi + 1$

$ci \leftarrow ci + 1$

C tömbbe tegyük a maradékot

Összefuttatás algoritmusa

- Összefuttat(A, B, C)

$ai \leftarrow 1,\ bi \leftarrow 1,\ ci \leftarrow 1$

$ai \le k \wedge bi \le m$

$A[ai] < B[bi]$	$A[ai] = B[bi]$	$A[ai] > B[bi]$
$C[ci] \leftarrow A[ai]$ $ai \leftarrow ai + 1$	$C[ci] \leftarrow A[ai]$ $ai \leftarrow ai + 1$ $ci \leftarrow ci + 1$ $C[ci] \leftarrow B[bi]$ $bi \leftarrow bi + 1$	$C[ci] \leftarrow B[bi]$ $bi \leftarrow bi + 1$

$ci \leftarrow ci + 1$

C tömbbe tegyük a maradékot

Összefuttatás algoritmusa

- Összefuttat(A, B, C)

$ai \leftarrow 1,\ bi \leftarrow 1,\ ci \leftarrow 1$

$ai \leq k \wedge bi \leq m$

$A[ai] < B[bi]$	$A[ai] = B[bi]$	$A[ai] > B[bi]$
$C[ci] \leftarrow A[ai]$ $ai \leftarrow ai + 1$	$C[ci] \leftarrow A[ai]$ $ai \leftarrow ai + 1$ $ci \leftarrow ci + 1$ $C[ci] \leftarrow B[bi]$ $bi \leftarrow bi + 1$	$C[ci] \leftarrow B[bi]$ $bi \leftarrow bi + 1$

$ci \leftarrow ci + 1$

C tömbbe tegyük a maradékot

Összefuttatás algoritmusa

- Összefuttat(A, B, C)

<table>
<tr><td colspan="4">$ai \leftarrow 1,\ bi \leftarrow 1,\ ci \leftarrow 1$</td></tr>
<tr><td rowspan="4"></td><td colspan="3">$ai \leq k \wedge bi \leq m$</td></tr>
<tr><td>$A[ai] < B[bi]$</td><td>$A[ai] = B[bi]$</td><td>$A[ai] > B[bi]$</td></tr>
<tr><td>$C[ci] \leftarrow A[ai]$
$ai \leftarrow ai + 1$</td><td>$C[ci] \leftarrow A[ai]$
$ai \leftarrow ai + 1$
$ci \leftarrow ci + 1$
$C[ci] \leftarrow B[bi]$
$bi \leftarrow bi + 1$</td><td>$C[ci] \leftarrow B[bi]$
$bi \leftarrow bi + 1$</td></tr>
<tr><td colspan="3">$ci \leftarrow ci + 1$</td></tr>
<tr><td colspan="4">C tömbbe tegyük a maradékot</td></tr>
</table>

Összefuttatás algoritmusa

- Összefuttat(A, B, C)

$ai \leftarrow 1,\ bi \leftarrow 1,\ ci \leftarrow 1$

$ai \leq k \wedge bi \leq m$

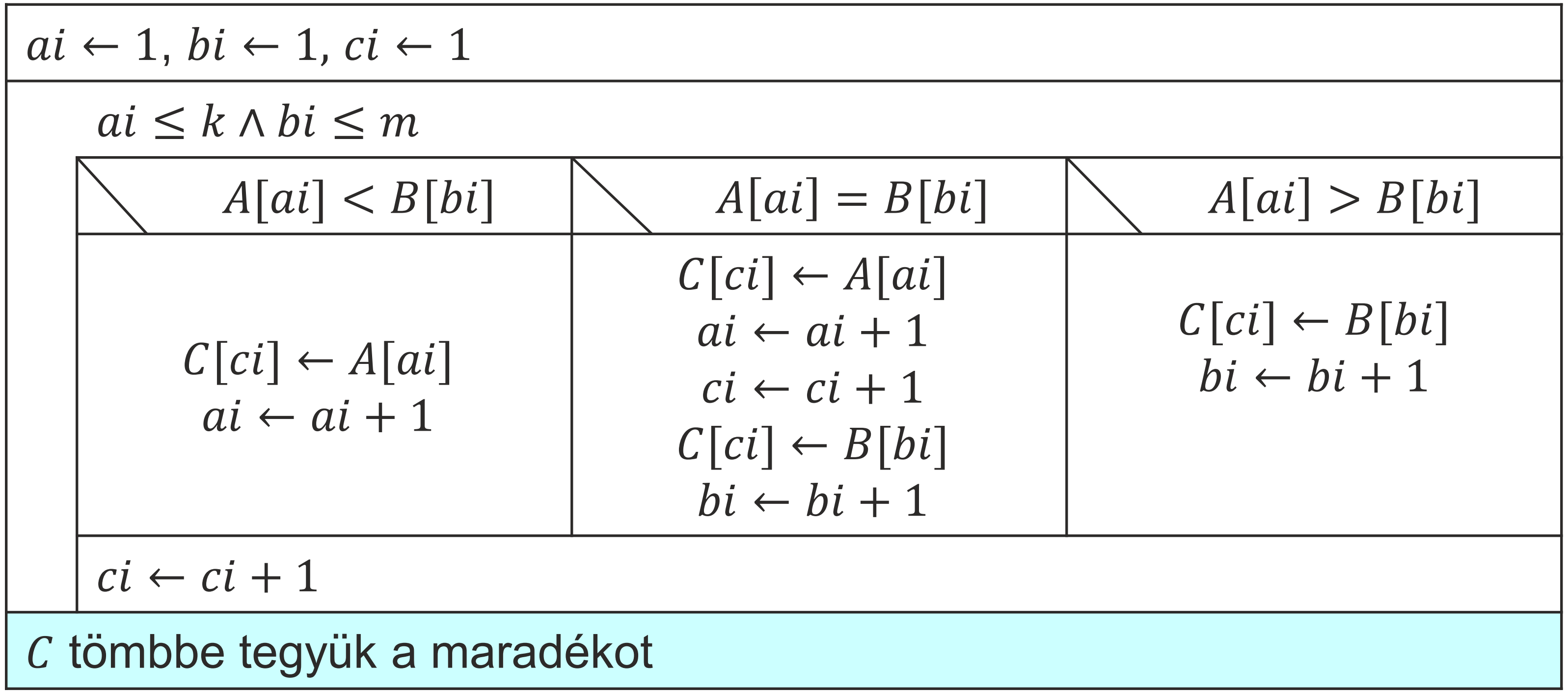
$A[ai] < B[bi]$	$A[ai] = B[bi]$	$A[ai] > B[bi]$
$C[ci] \leftarrow A[ai]$ $ai \leftarrow ai + 1$	$C[ci] \leftarrow A[ai]$ $ai \leftarrow ai + 1$ $ci \leftarrow ci + 1$ $C[ci] \leftarrow B[bi]$ $bi \leftarrow bi + 1$	$C[ci] \leftarrow B[bi]$ $bi \leftarrow bi + 1$

$ci \leftarrow ci + 1$

C tömbbe tegyük a maradékot

Összefuttatás algoritmusa

- Összefuttat(A, B, C)



- Ez egy külön ciklus

Összefuttatásos rendezés

- Összefuttatásnál
  - összehasonlítások száma: $k + m - 1$
- RENDEZÉS:
  - Az üres és az egyelemű sorozat biztosan rendezett.
  - $S[1..n]$ $(n > 1)$ tömböt ha rendezni szeretnénk:
    - szétvágjuk két részre (S_1, S_2) – elfelezzük
    - ezeket külön-külön rendezzük
    - összefuttatjuk
  - MergeSort-nak is hívják

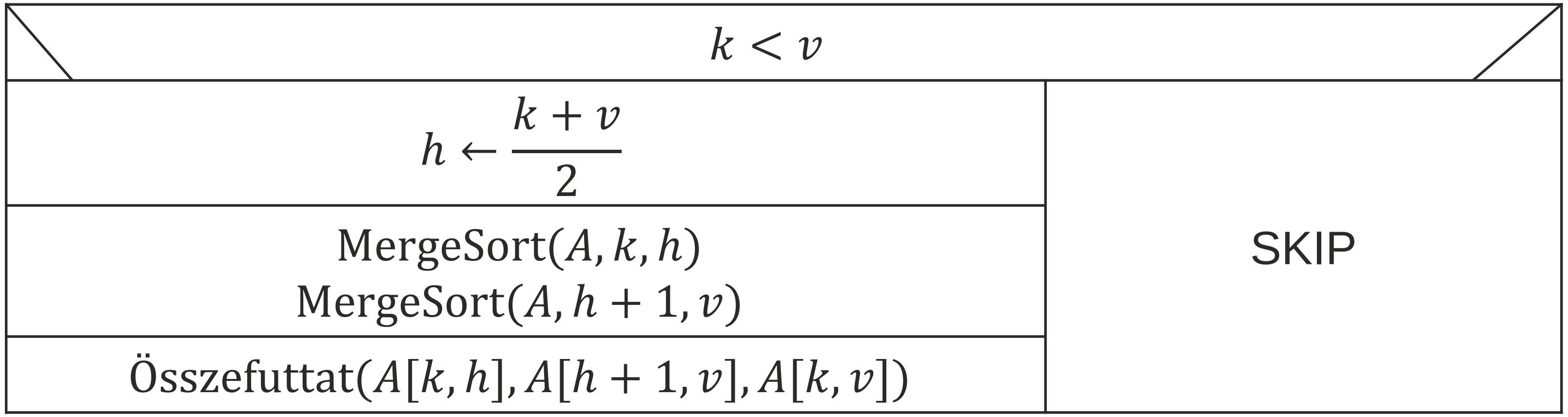
Összefuttatásos rendezés

- $\text{MergeSort}(S)$

$\text{Length}(S) > 1$	
$\text{Szétvág}(S, S_1, S_2)$	SKIP
$\text{MergeSort}(S_1)$ $\text{MergeSort}(S_2)$	
$\text{Összefuttat}(S_1, S_2, S)$	

Összefuttatásos rendezés – tömbre

- $\text{MergeSort}(A, k, v)$



- A külső hívás
  - $\text{MergeSort}(A, 1, n)$
- Az összefuttatásnál szükséges egy segédtömb használata

Összefuttatásos rendezés

- Tömbökre szokás egy másik, iteratív stratégiát is követni:
  - Minden lépésben végigmegyünk rajta egyre növekvő $h = 1, 2, 2^2, 2^3, \dots\ 2^{[\log_2 n]-1}$ lépésközzel, és az ilyen hosszú szomszédos tömbrészleteket fésüljük össze, egy segédtömböt használva
- Példa …

Összefuttatásos rendezés

8	2	3	11	4	5	9	7	1	6	10

2	8	3	11	4	5	7	9	1	6	10

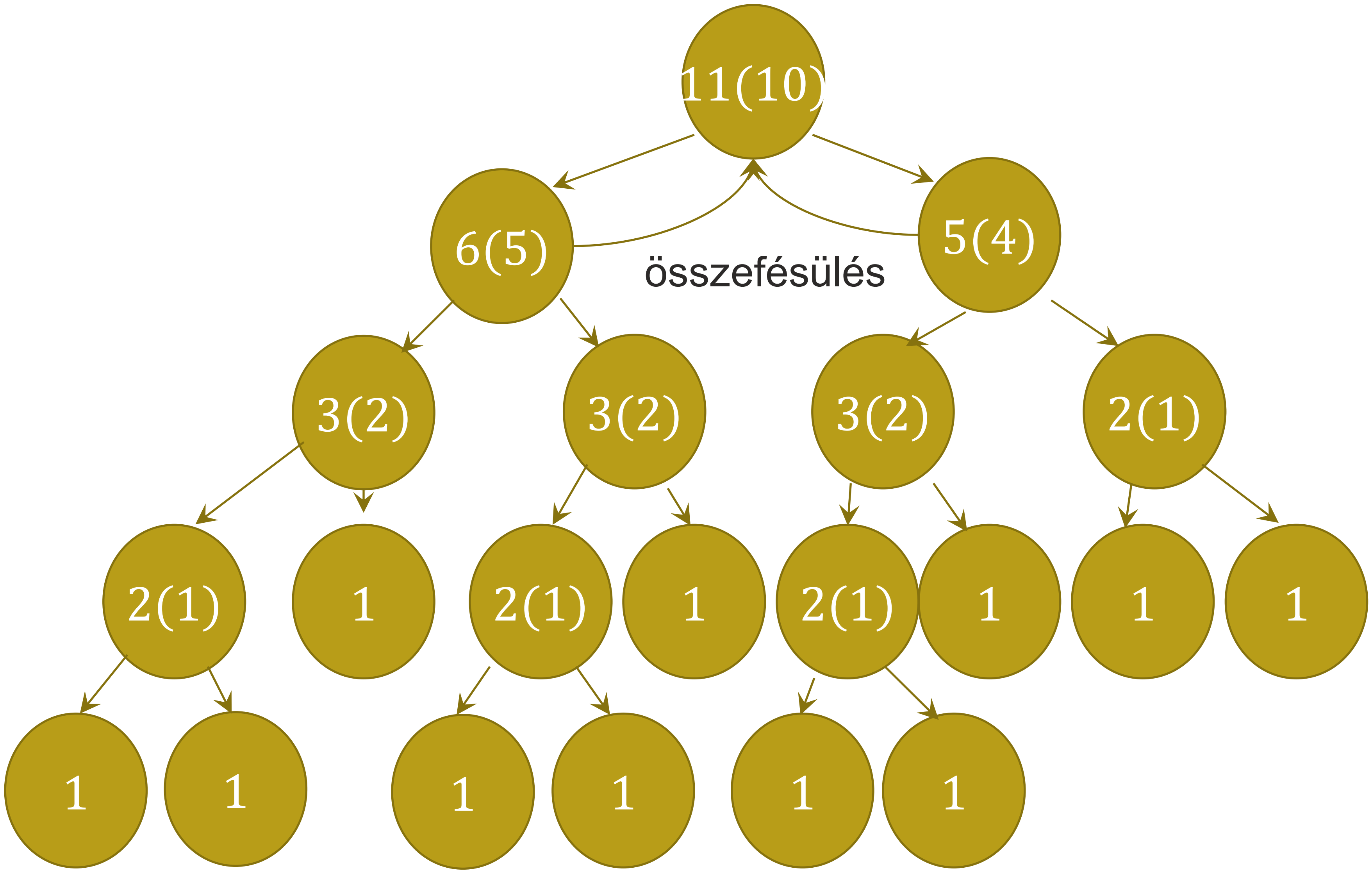
2	3	8	11	4	5	7	9	1	6	10

2	3	4	5	7	8	9	11	1	6	10

1	2	3	4	5	6	7	8	9	10	11

Összefuttatásos rendezés

- Hatékonyságelemzés:
  - Az összehasonlítások számát becsüljük a legrosszabb esetben
    - Az összefuttatás k és m hosszú sorozatok összefésülésekor $k+m-1$ összehasonlítást végez
      - $\mathrm{MÖ}_{\text{összefuttat}}(k,m) = k+m-1$
    - ha n a két sorozat együttes hossza
      - $\mathrm{MÖ}_{\text{összefuttat}}(n) = n-1$
    - A MergeSort eljárásnál a közel egyenlő szétvágást alapul véve: milyen hosszú részekre vágja az eljárás a (rész)sorozatokat a rekurzív hívások szintjein.



$n = 11$-re

( )-ben:öh-k száma $\text{MÖ} = 10 + (5 + 4) + (2 + 2 + 2 + 1) + (1 + 1 + 1) = 29$

Összefuttatásos rendezés

- 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 $\mathrm{MÖ}_{\mathrm{MS}}(n) \leq (n-1) * \lceil \log_2 n \rceil = \Theta(n * \log_2 n)$

Batcher-féle rendező

Következő téma