CS 4800: Algorithms & Data

Lecture 3

January 16, 2018

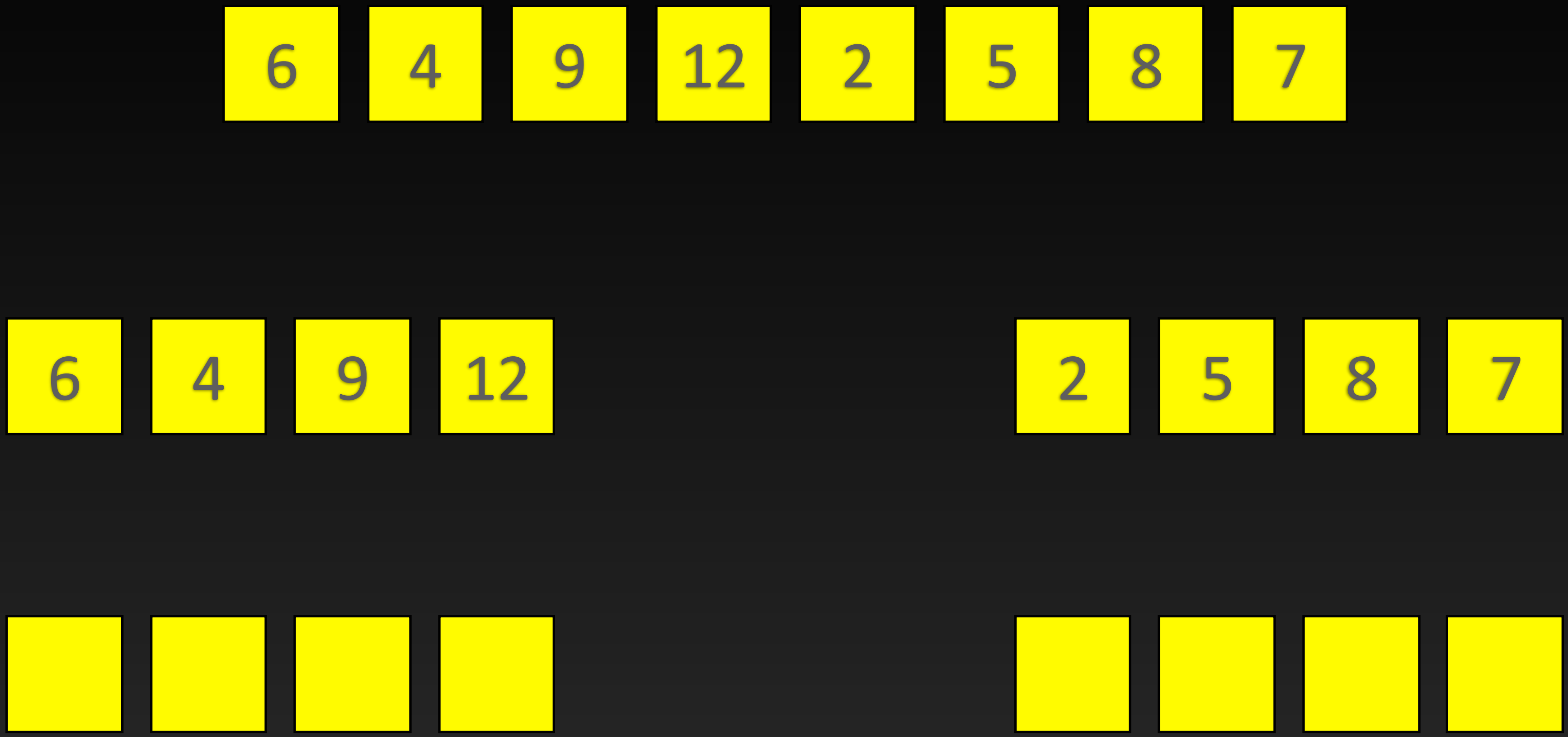
Mergesort

- A[1…n] : n numbers
- Sort A in non-decreasing order using divide-and-conquer

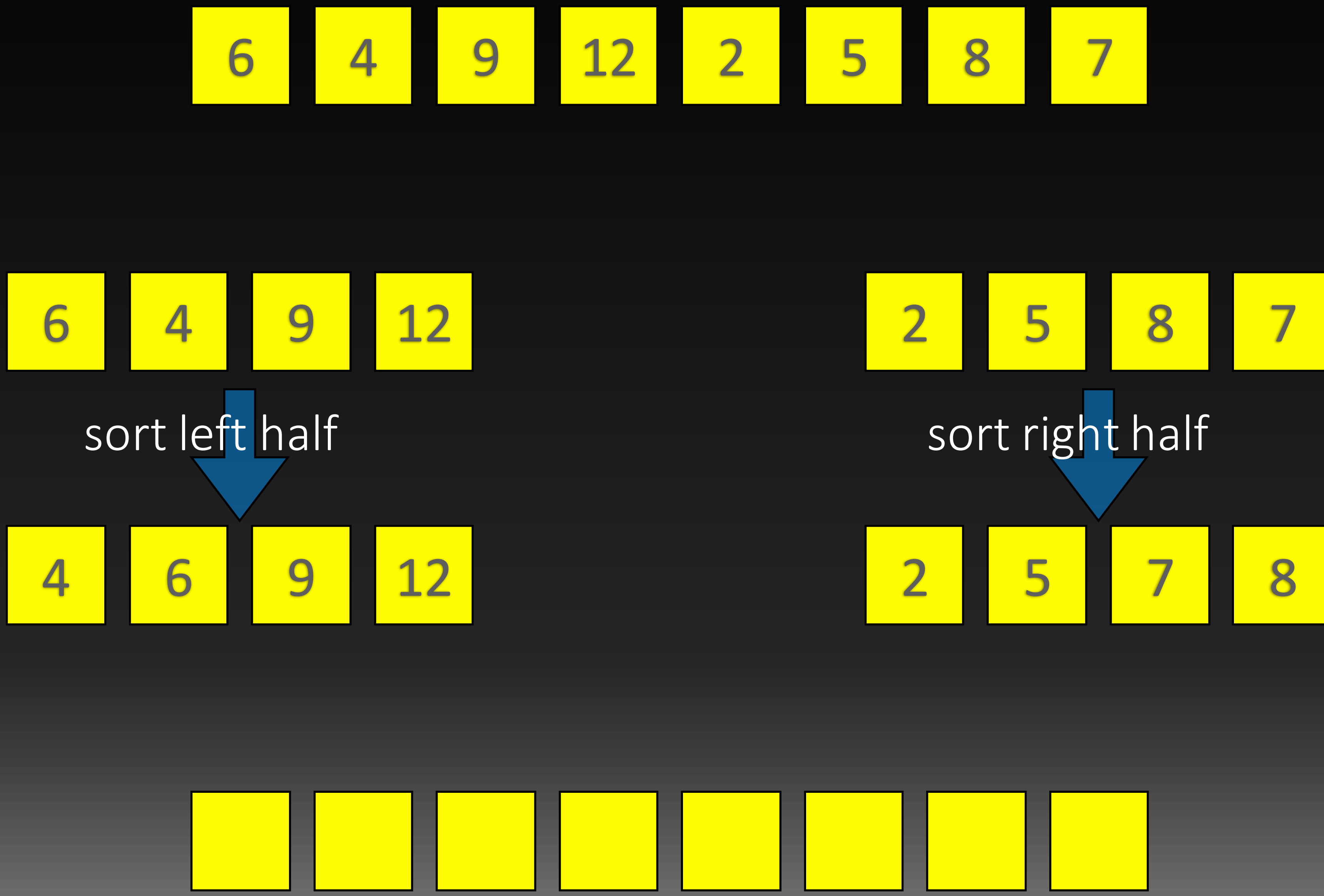
Mergesort

6	4	9	12	2	5	8	7

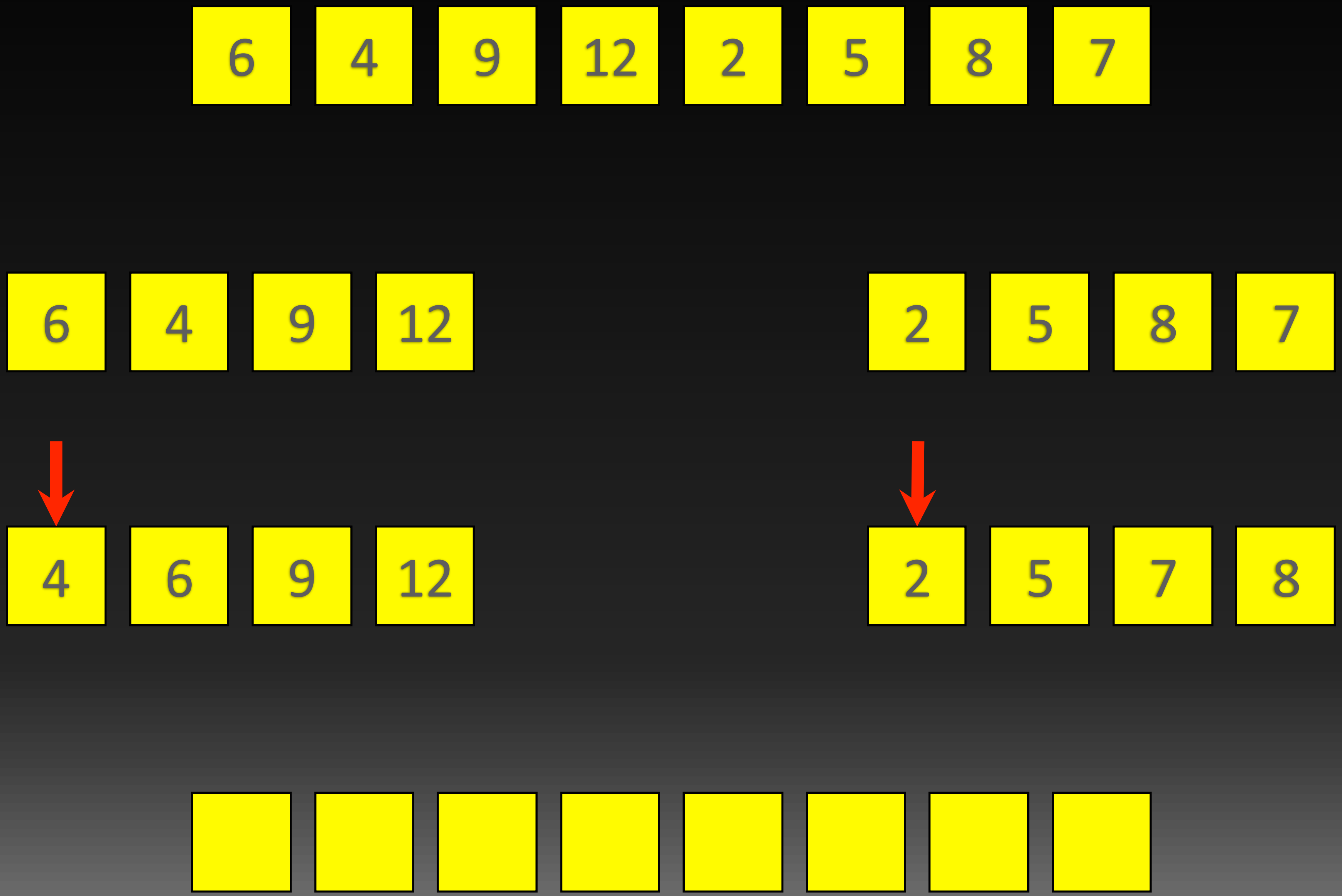
Mergesort



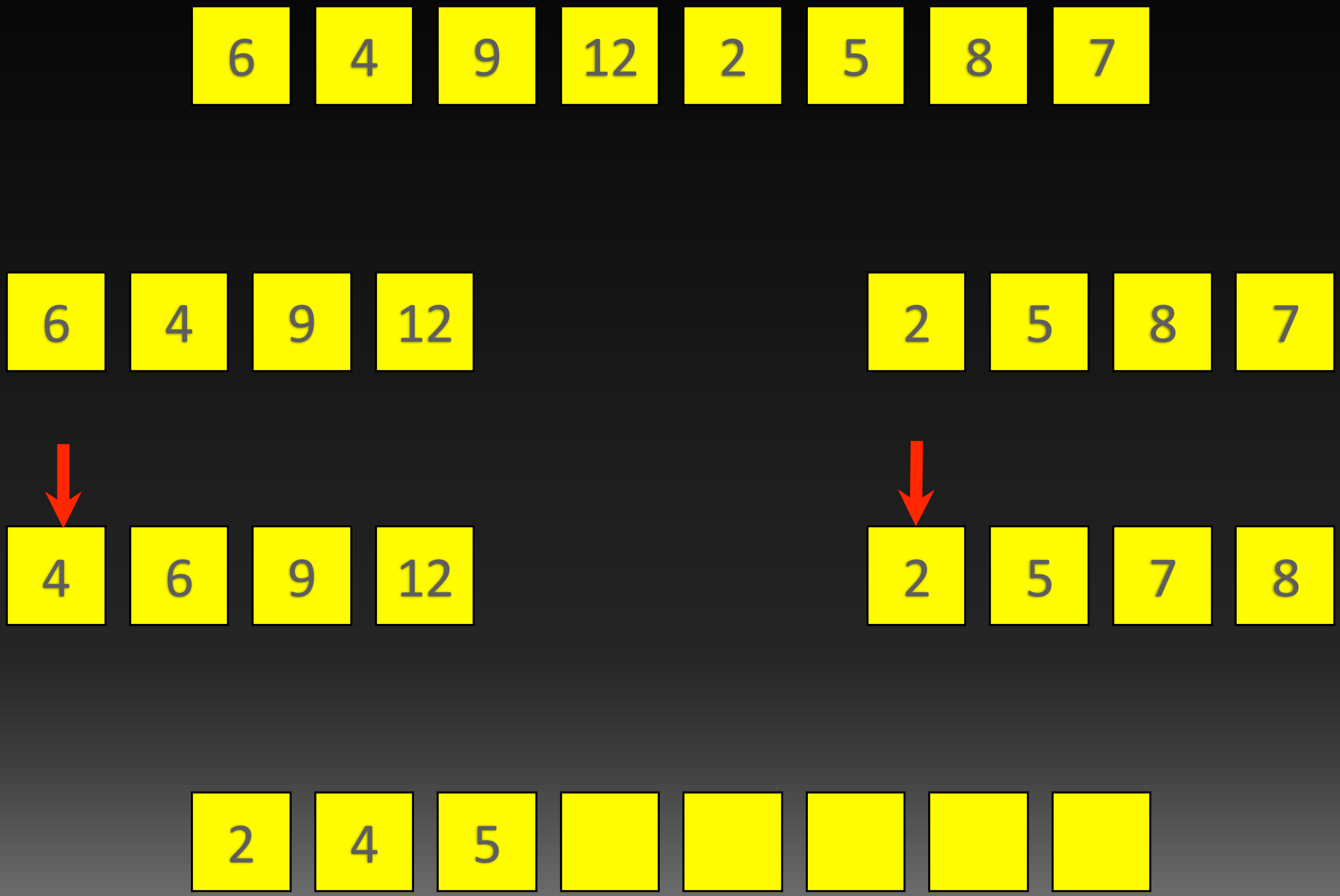
Mergesort


6 4 9 12 2 5 8 7
6 4 9 12
2 5 8 7
sort left half
sort right half
4 6 9 12
2 5 7 8

Mergesort



Mergesort



merge-sort(A, l, r) (sort A[l…r])

If l < r

$mid = \lfloor (l + r) / 2 \rfloor$

merge-sort(A, l, mid) — Subproblem of size n/2

merge-sort(A, mid+1, r) — Subproblem of size n/2

merge(A, l, mid, r) — O(n) work

merge(A, l, mid, r)

$aux[l, \ldots, mid] \leftarrow a[l, \ldots, mid]$

$aux[mid + 1, \ldots, r] \leftarrow a[r, \ldots, mid + 1]$

$i \leftarrow l, j \leftarrow r$

For $k \leftarrow l\ to\ r$

if $aux[i] < aux[j]$ then

$a[k] \leftarrow aux[i]$

$i \leftarrow i + 1$

else

$a[k] \leftarrow aux[j]$

$j \leftarrow j - 1$

Sedgewick

$$T(n) = 2T(n/2) + cn$$

prove: $T(n) = O(n \log n)$

hypothesis: $T(n) \leq cn\,(1 + \log_2 n)$

base case: $T(1) \leq c$

inductive step:

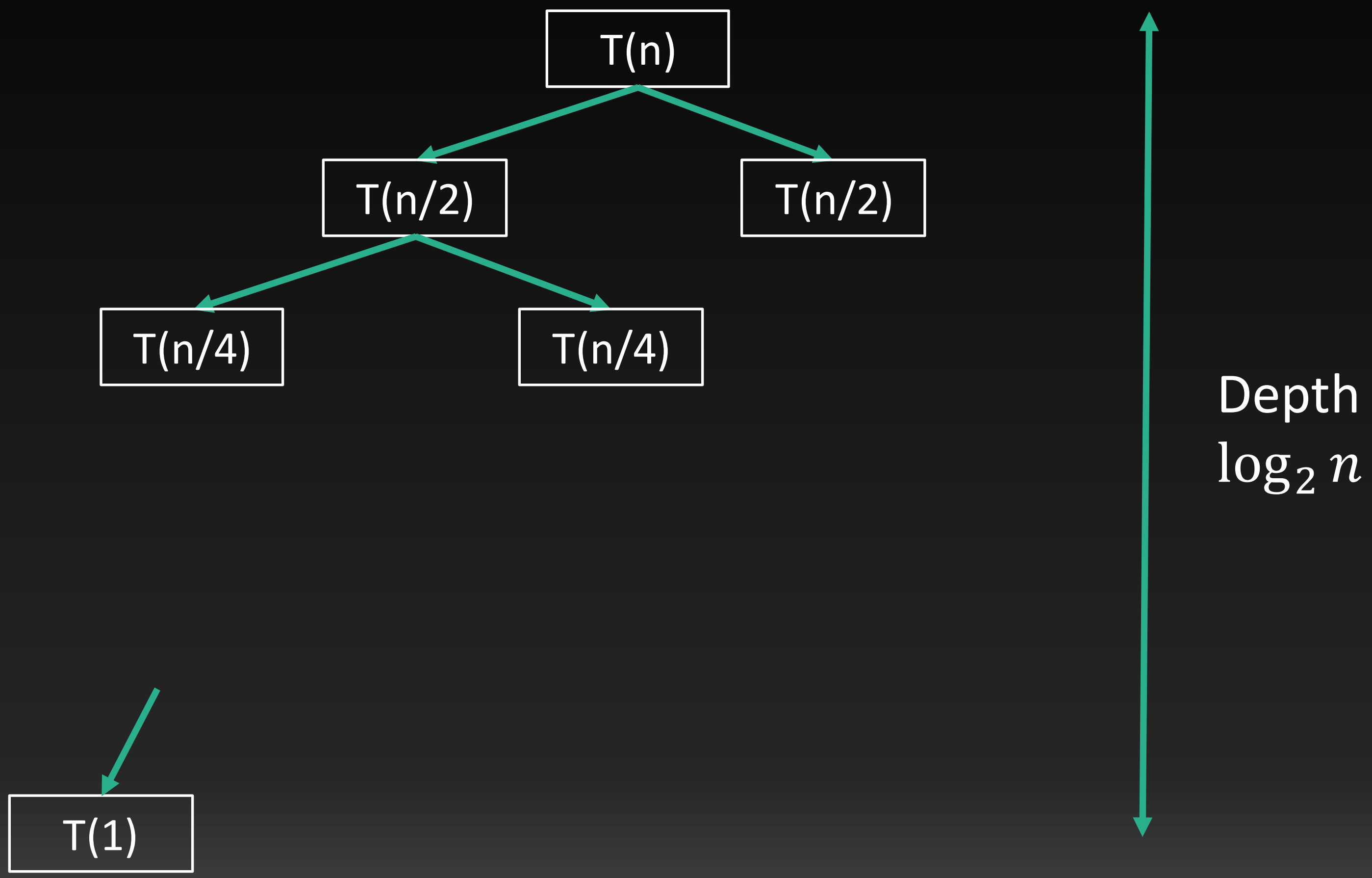
$$T(k) = 2T\left(\frac{k}{2}\right) + ck$$

$$T(k) \leq 2c\frac{k}{2}(1 + \log_2(k/2)) + ck$$

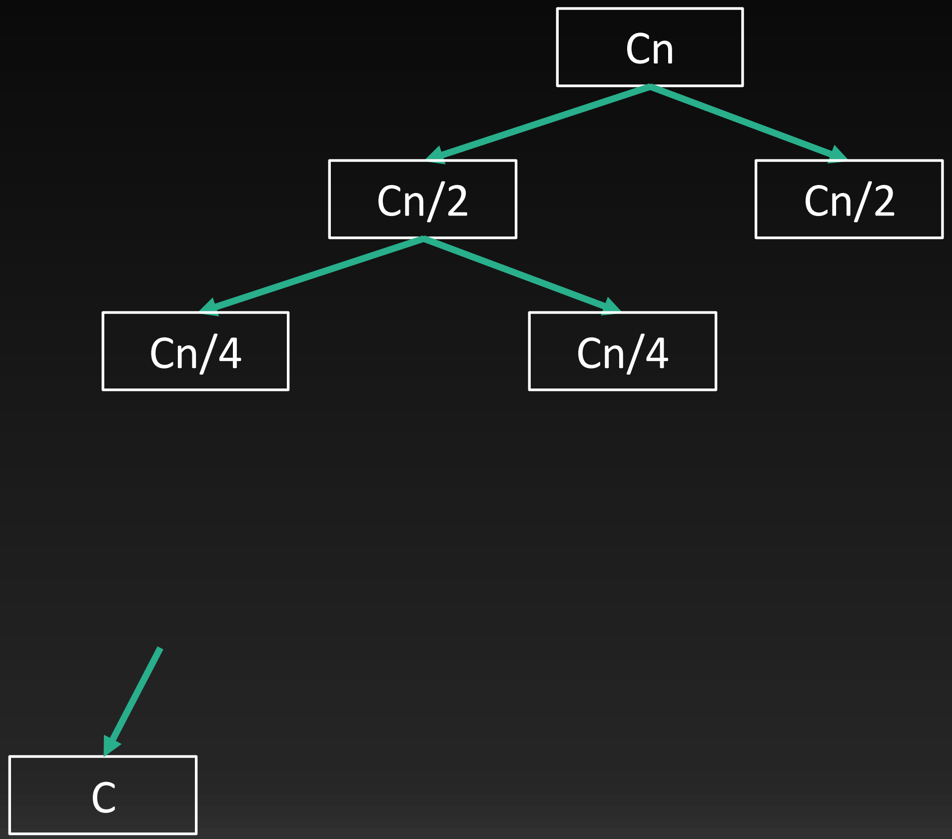
$$T(k) \leq ck\,(1 + \log_2 k - 1) + ck$$

$$T(k) \leq ck\,(1 + \log_2 k)$$

Recursion tree



Amount of work



Cn

2Cn/2 = Cn

4Cn/4 = Cn

$2^{\log_2 n} Cn/2^{\log_2 n}$

Total running time = sum over all merging time

Summing over all levels

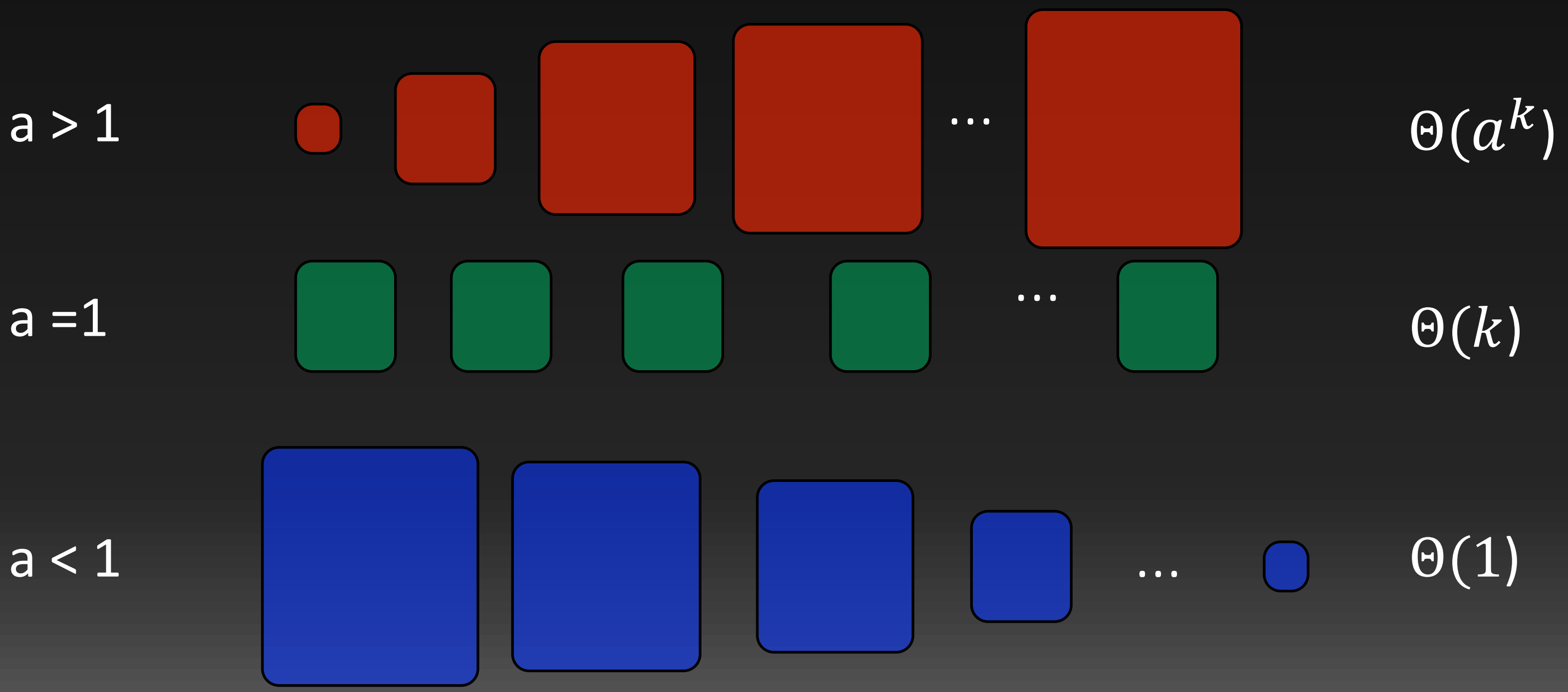
- $\log_2 n$ levels
- Running time of each level: Cn
- Total running time: $Cn(1 + \log_2 n)$

Geometric series

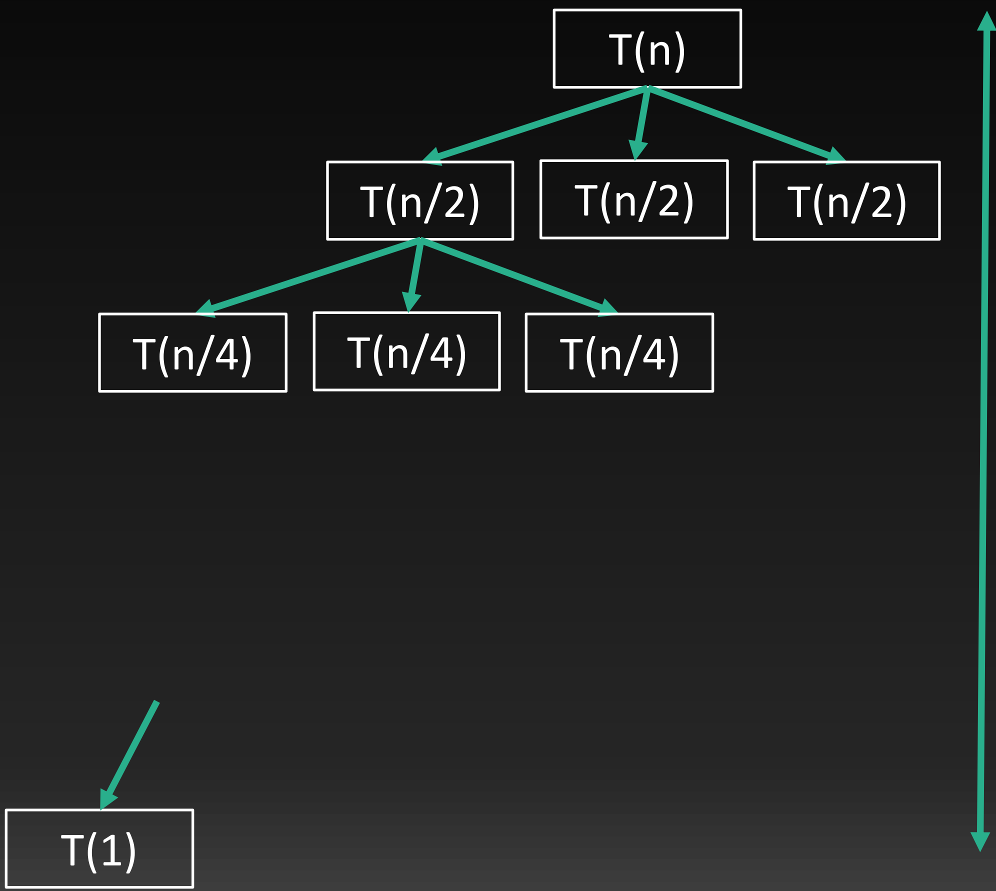
- $S = 1 + a + a^2 + a^3 + \cdots + a^k = ?$
- $aS = \qquad a + a^2 + a^3 + \cdots + a^k + a^{k+1}$
- $aS - S = a^{k+1} - 1$
- $S = \frac{a^{k+1}-1}{a-1}$

Geometric series

- $S = 1 + a + a^2 + a^3 + \cdots + a^k = \frac{a^{k+1}-1}{a-1}$



Karatsuba

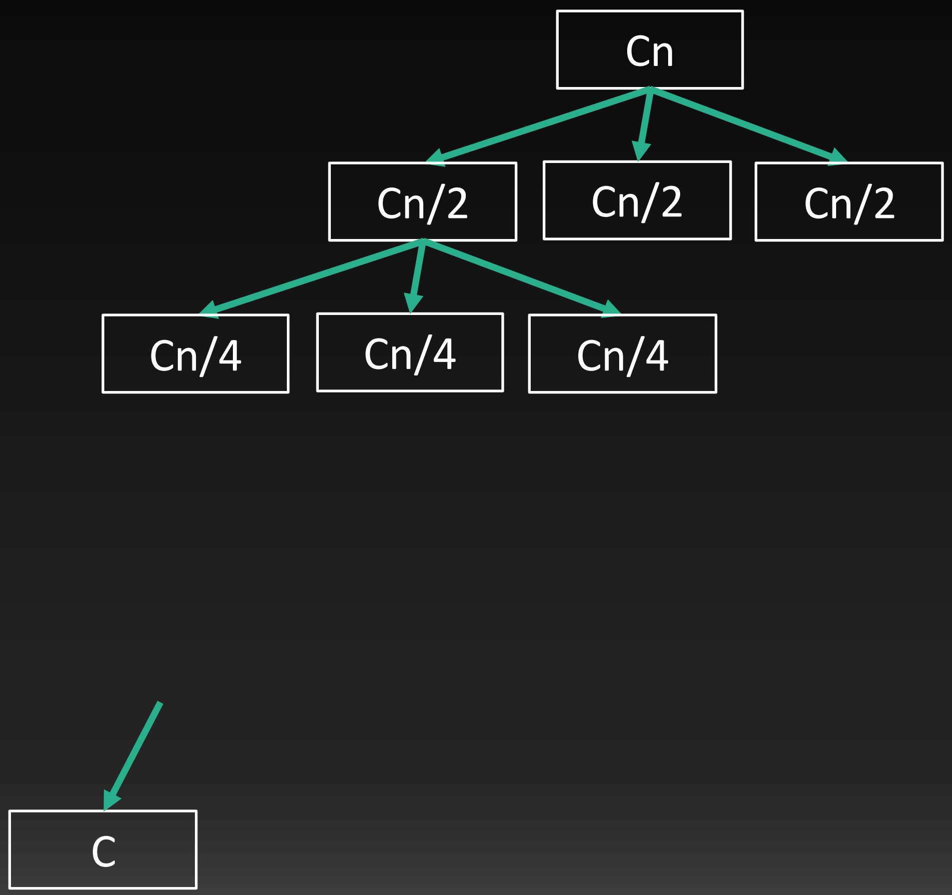


Karatsuba(X, Y, n)

- If n = 1 then return $X \cdot Y$
- Else:
  - $m = \lceil n/2 \rceil$
  - Rewrite $X = 10^{\lceil n/2 \rceil} a + b$
  - $Y = 10^{\lceil n/2 \rceil} c + d$
  - $e = Karatsuba(a, c, m)$
  - $f = Karatsuba(b, d, m)$
  - $g = Karatsuba(a - b, c - d, m)$
  - Return $10^{2m} e + 10^{m}(e + f - g) + f$

Depth
$\log_2 n$

Additional work



Cn

3Cn/2

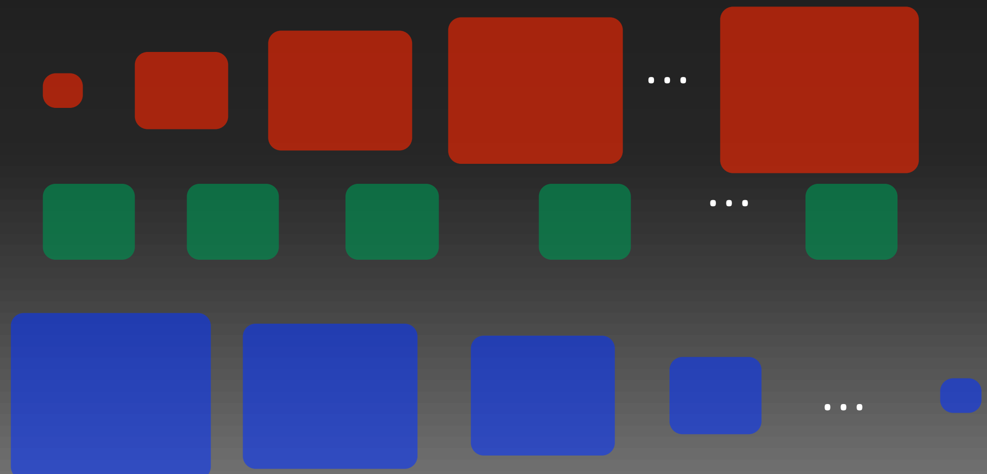
9Cn/4 = $(3/2)^2$ Cn

$(3/2)^{\log_2 n} Cn$

Summing over all levels

- $\log_2 n$ levels
- Running time of each level:
  - Cn, $(3/2)^1$ Cn, $(3/2)^2$ Cn, …, $(3/2)^{\log_2 n} Cn$
- Total running time:
- $Cn + \left(\frac{3}{2}\right)^1 Cn + \cdots + \left(\frac{3}{2}\right)^{\log_2 n} Cn = ?$

$$S = 1 + a + a^2 + a^3 + \cdots + a^k = \frac{a^{k+1} - 1}{a - 1}$$



a > 1 $\Theta(a^k)$

a = 1 $\Theta(k)$

a < 1 $\Theta(1)$

Tricks with log

- $a^{\log_b c} = c^{\log_b a}$. How to prove?
- Take the log base b of both sides
- $\log_b a \log_b c = \log_b c \log_b a$
- Running time of Karatsuba
    - $C \cdot 3^{\log_2 n}$
    - $C \cdot n^{\log_2 3}$
    - The same!