Mergesort

↳ Merge Procedure

↳ combining talk

[50, 70, 45, 5, 79, 47, 29, 52]

0 1 2 3 4 5 6 7

$n = 8$

Small Problem

↳ single element

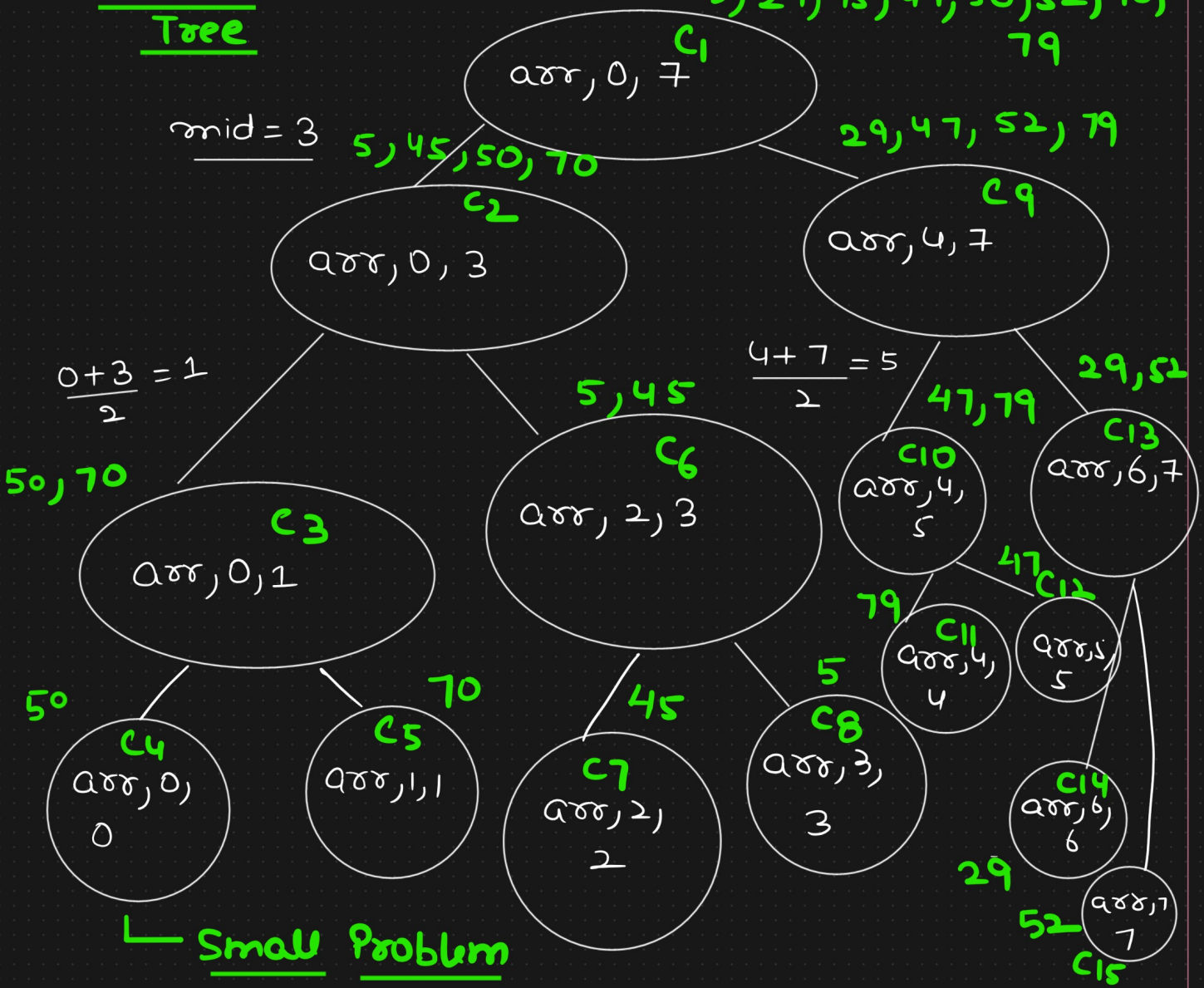
↳ sorted array

Big Problem $n > 1$

Sorted array

5, 29, 45, 47, 50, 52, 70, 79

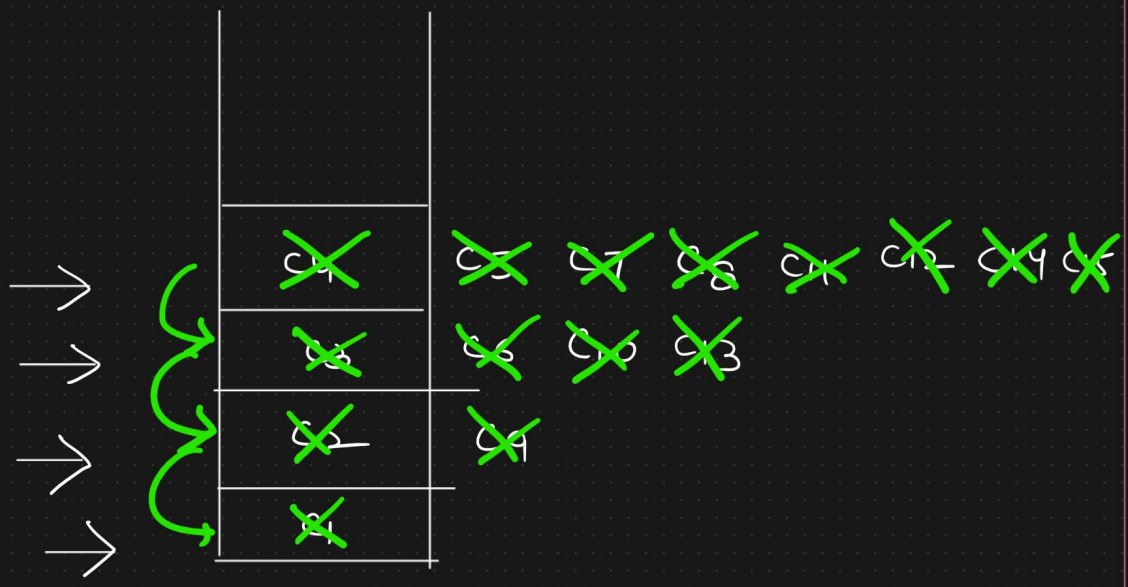
Recursive Tree



↳ Small Problem

$i == j \Rightarrow$ array contains only single element

Stack data structure $= O(\log_2 n)$



Top to Down ⟶ Divide & conquer

↳ Small problem

Down to Top ⟶ combine

↳ Merge Procedure

↓↓

fully sorted

subarray

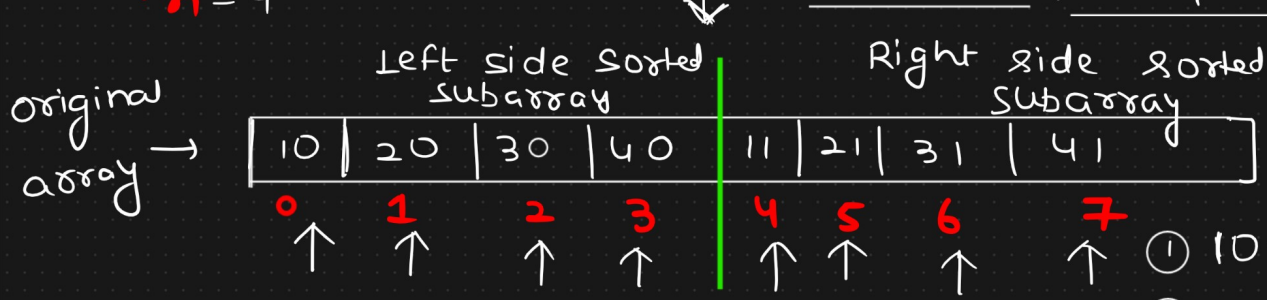
Worst case scenario

Time complexity of merge procedure

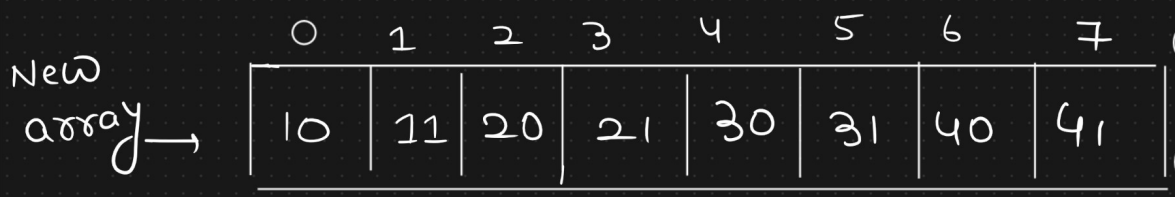
↳ Number of moves

Number of comparisons

$n = 4$ $m = 4$



Merge Procedure

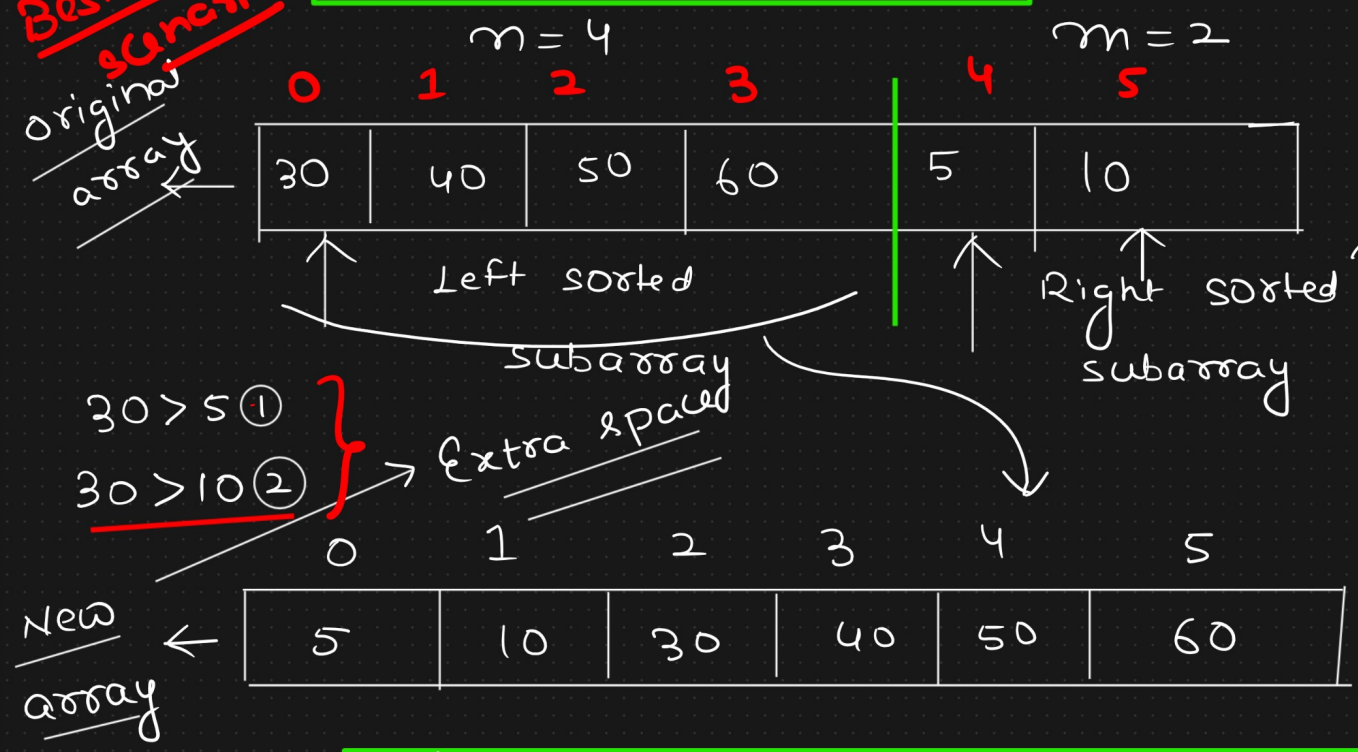


① $10 < 11$
② $20 > 11$
③ $20 < 21$
④ $30 > 21$
⑤ $30 < 31$
⑥ $40 > 31$
⑦ $40 < 41$

Worst case scenario

↳ $n + m - 1$

Best case scenario



Best case scenario

comparisons = $\min(n, m)$

$= \min(2, 4) = 2$

moves = $m+n$

$m+n = N$

Time complexity → $O(m+n)$

(Merge Procedure) → $O(N)$

⇓

moves + # comparisons

$(m+n) + (m+n-1)$

$\Rightarrow O(m+n)$

Space complexity → $O(N)$

⇓

extra space (new array)

⇓

Outplace sorting algorithm

$T(n) \Rightarrow$ mergeSort(arr, i, j):

Psuedocode

if i == j: } c — small Problem

return arr

else:

c — Divide → mid = i + (j-i)/2

$2T(N/2)$ — Conquer → $T(N/2)$ mergeSort(arr, i, mid)

$T(N/2)$ mergeSort(arr, mid+1, j)

$O(N)$ — Combine → mergeProcedure(arr, i, mid, mid+1, j)

return arr

Recurrence Relation

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

- Master's Theorem
- Substitution Method

$a = 2$ $k = 1$

$b = 2$ $p = 0$

Masters Theorem

$$\log_b^a = \log_2^2 = 1$$

$$\log_b^a = k$$

(1) (1)

$$\Rightarrow \Theta(N \log N)$$

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

$$= 2\left(2T(n/2^2) + n/2\right) + n$$

Substitution Method

$$\Rightarrow 2^2 T(n/2^2) + 2n$$

$$\Rightarrow 2^2\left(2T(n/2^3) + \frac{n}{2^2}\right) + 2n$$

$$\Rightarrow 2^3 T\left(\frac{n}{2^3}\right) + 3n$$

$$\frac{n}{2^k} = 1 \qquad \downarrow k$$

$$k = \log_2 n$$

$$2^k T\left(\frac{n}{2^k}\right) + k \cdot n$$

$$2^{\log_2 n} T\left(\frac{n}{2^{\log_2 n}}\right) + \log_2 n \cdot n$$

$$n^{\log_2 2} T\left(\frac{n}{n^{\log_2 2}}\right) + n \log_2 n$$

$$n + n \log_2 n$$

$\Rightarrow O(m \log n)$