

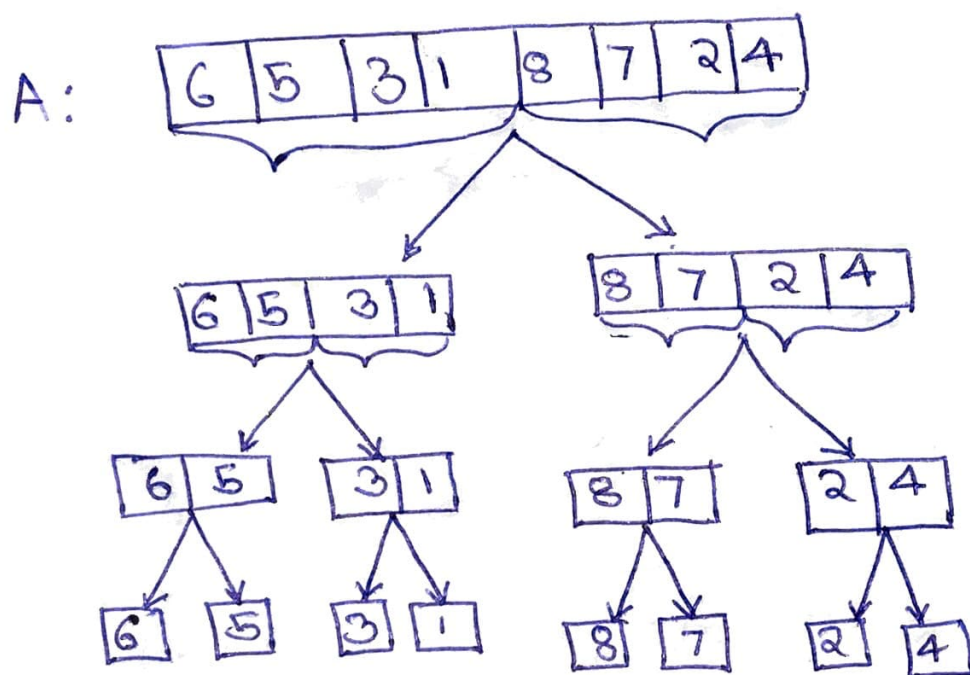
"Merge sort"

1) Insertion sort \rightarrow Worst case $O(n^2)$.
Best case $O(n)$.

2) Merge sort \rightarrow Worst case $O(n \log n)$.

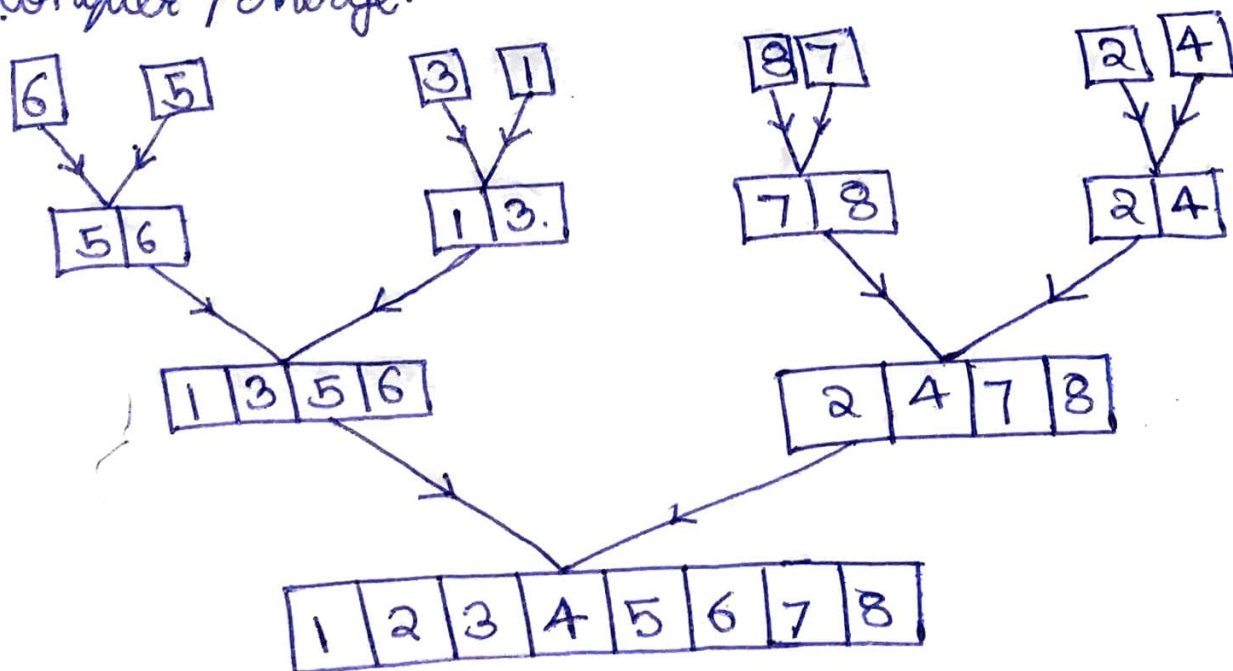
a) Divide and Conquer algorithm

b) Large amount of data \rightarrow can't store in RAM.
so stored in Hard disk or SSD.



i) DIVIDE.

★ ii) Conquer / Merge.



★ Pseudo code for Merge sort.

\Rightarrow Merge-Sort (A, p, r) \rightarrow index of last element
if $p < r$ \rightarrow index of first element
 $m = \lfloor (p+r)/2 \rfloor \rightarrow$ floor.

$$q = \lfloor L(p+x)/2 \rfloor \rightarrow \text{floor.}$$

② $\Gamma \rightarrow \underline{\underline{\text{ceil}}}$.

(Recursion)

Merge-Sort (A, p, q)

Merge-sort ($A, q+1, r$)

Merge (A, p, q, x)

$\Rightarrow \text{Merge}(A, p, q, r)$

Ex: $q, q+1, \dots, r$

p								r
6	5	3	1	8	7	2	4	
1	2	3	4	5	6	7	8	

$$C \leftarrow \begin{cases} n_1 \leftarrow q - p + 1 & \Rightarrow 4 - 1 + 1 \Rightarrow \underline{4} \\ n_2 \leftarrow x - q & \Rightarrow 8 - 4 \Rightarrow \underline{4} \end{cases}$$

```
// create arrays L[1...n1+1] and R[1...n2+1]
```

$C.n/2 \leftarrow \begin{bmatrix} \text{for } i \leftarrow 1 \text{ to } n, \\ \text{do } L[i] \leftarrow A[p+i-1] \end{bmatrix}$

```

c.n/2 ← [ for j ← 1 to n₂
           do R[j] ← A [q + j]

```

$$c \leftarrow f_2[n_1 + 1] \leftarrow \infty$$
$$R[n_2+1] \leftarrow \infty$$

↓

$$l_j \leftarrow 1$$

Time $\rightarrow O(n)$

space $\rightarrow O(n)$

for $k \leftarrow p$ to x

do. if $L[i] \leq R[j]$

then $A[K] \leftarrow L[i]$

$$i \leftarrow i + 1$$

doe $A[k] \leftarrow R[j]$

$$j \leftarrow j + 1$$

C. n

$$T(n) = \underbrace{2 \times T\left(\frac{n}{2}\right)}_{\substack{\downarrow \\ 2 \times T\left(\frac{n}{4}\right) + T\left(\text{Merge array of size } \frac{n}{4}\right) \\ \downarrow \\ 2 \times T\left(\frac{n}{8}\right) + T\left(\text{Merge array of size } \frac{n}{8}\right)}} + T(\text{Merge arrays of size } \frac{n}{2})$$

$O(n)$
 $O(n/2)$
 $O(n/4)$

Time

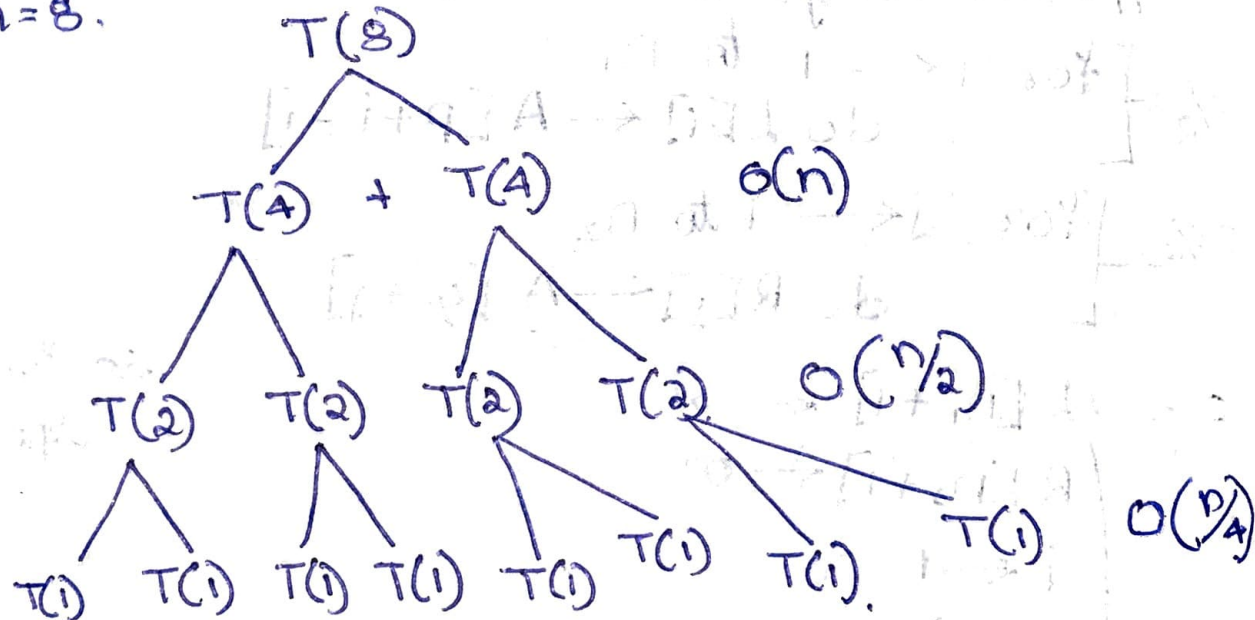
$$T(n) = T(n/2) + T(n/2) + O(n)$$

$$T(n) = 2T(n/2) + O(n) \rightarrow \text{recurrence - Eq/Relation}$$

$$Space(n) = C + O(n)$$

Discrete Maths

Let $n=8$.



final $T(n) = 2 \times T(n/2) + O(n)$

$$S(n) = O(n)$$