WiFi

Students Network

Username: guest

Password: h1SiPCoT

How long does this take?

SIM card list size S

Aadhaar list size A

$S \times A$

for each SIM card

? Unordered

Worst Case

Scan entire list

Aadhaar

SIM Cards

Which SIM Cards do not have corresponding Aadhaar

Match name/address

For every SIM card

Search for matching entry in Aadhaar data

$S = ?$ 10^9 1 billion 100 crore

$A = ?$ 10^9

$S \times A = 10^{18}$

Speed of a desktop computer?

10^9 ops/sec

Time for $S \times A = \frac{10^{18}}{10^9} = 10^9$ sec

60 sec → min

60 × 60 — 3600 sec hour

× 24 day

≈ 100,000 sec / day 10^5

10^8 sec ≈ 3 years

10^9 sec ≈ 30 years

Arrange Aadhaar data systematically

Sort alphabetically by Name, Address

1000 numbers $(N/2)/2$

500

250

$2^k = N$

$2^{10} = 1024$ $k = \log_2 N$

Binary search on A $\simeq 10^9$ entries

$\log_2 10^9 = ?$ $\Longrightarrow$ 30

$1024 = 10^3$ $\log_2 1000 \simeq 10$

$10^9 = 10^3 \cdot 10^3 \cdot 10^3$

$2^{30} = 2^{10} \cdot 2^{10} \cdot 2^{10}$

$S \times \log_2 A$

$10^9 \times 30 = 3 \times 10^{10}$

$30 \times \frac{10^9}{\text{sec}}$

$S \approx A \simeq N$

Brute force: N^2 = 30 yrs

Clever version: $N \log N$ = 30 secs

10^9 ops/sec

Complexity	Feasible Input Size
$\log N$	2^{10^9}
N	10^9
$N \log N$	$10^8 \sim 10^9$
N^2	10^4
N^3	10^3

Insertion Sort

Choose max ⟶ 1st

Choose 2nd max ⟶ 2nd

Select

Selection Sort

Sorting

Graded exam papers

Arrange in order of marks

Start with one

Position second relative to first

⋮

Position k^{th} paper wrt $1, 2, \ldots k-1$

Insert k^{th} paper in sorted $1 \ldots k-1$

Time?

Insertion Sort — N items

$N \times$

$0 + 1 + 2 + \cdots + (N-1)$

$N-1 + N-2 + \cdots + 0$

$= \frac{N \cdot (N-1)}{2} \approx N^2$

$O(N^2)$. For all N, $f(N) \leq cN^2$ for some c

Selection Sort

N passes – find the max of N, N-1, N-2, ... 1

$N + N-1 + \cdots + 2 + 1$

$\frac{N \cdot N+1}{2} \simeq O(N^2)$

Selection Sort – Always N^2

Insertion Sort – Varies according to input

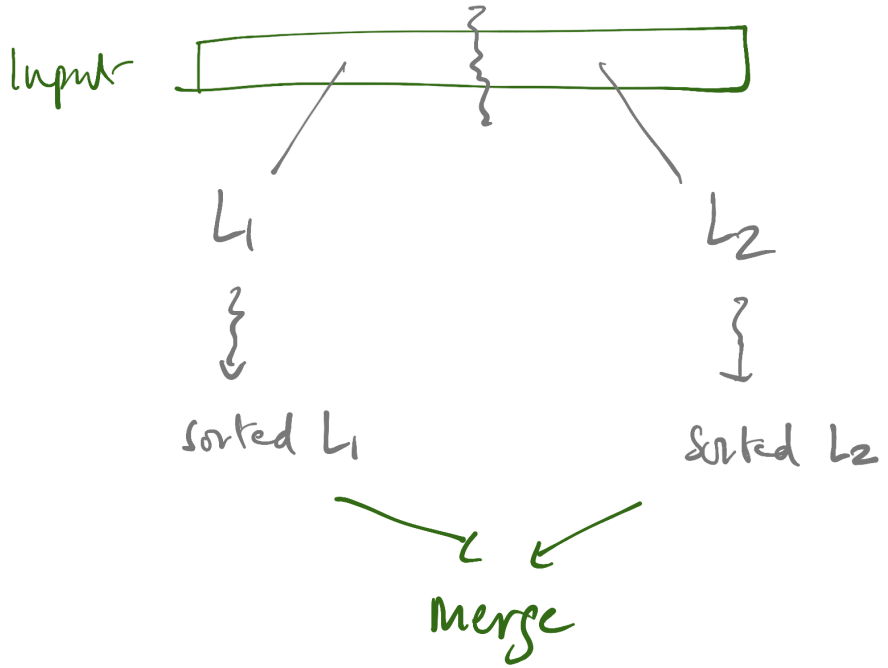
Ideally – "average case"

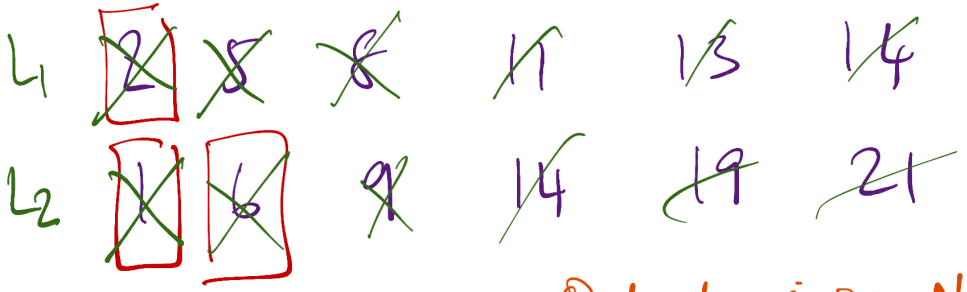
S, A $\longrightarrow$ A^2 insertion sort

$\downarrow$

S x log A search

Divide & Conquer

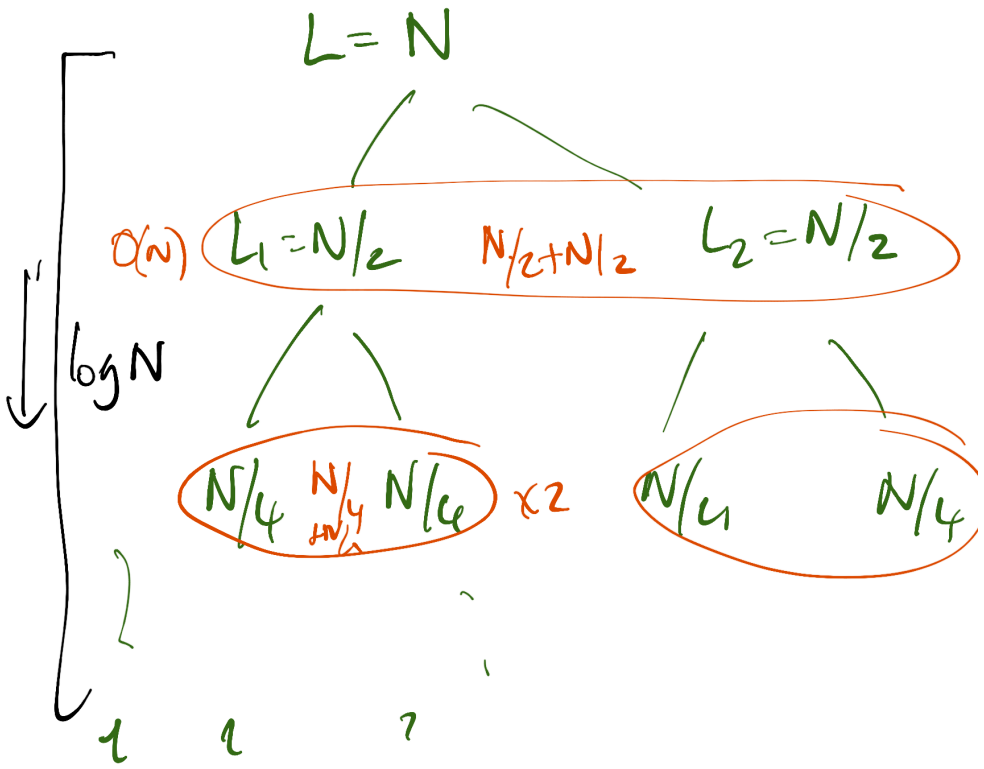




$O(N)$ time – Output is size N
One comparison per output element

1 2 5 6 8 9 11 13

14 14 19 21



Each level requires $O(N)$ work to merge

$O(N)$

$O(N/2) \times 2$

$O(N/4) \times 4$

Overall $O(N \log N)$

Binary Search

$$T(N) = 1 + T(N/2)$$
$$T(1) = 1$$

$T(N) = 1 + T(N/2)$

$= 1 + 1 + T(N/4)$

$= 1 + 1 + 1 + T(N/8)$ --

$= 1 + 1 + 1 + \dots T(N/N)$

log

Insertion Sort

$$T(N) = T(N-1) + N$$

$T(N-2) + N-1$

$T(N-3) + N-2$

$T(1) + 2$

$T(0) + 1$

Similarly Selection Sort

Merge Sort

$$T(N) = 2 \cdot T(N/2) + N$$

$$= 2 \cdot \left(2 \cdot T(N/4) + N/2\right) + N$$

$$= 2^2 \, T(N/2^2) + 2N$$

$$\vdots$$

$$= 2^k \, T(N/2^k) + kN$$

$k = \log N$

$$2^k \, T(N/2^k) + kN, \quad k = \log N$$

$N \quad T(1) \quad (\log N) \cdot N$

$$N + N \log N$$