

Linear search

12 August 2023 13:13

arr = [3, 5, 1, 2, 7]
0 1 2 3 4
~~i~~ ~~i~~ ~~i~~ ~~i~~ ~~i~~
for (i = 0; i < n; i++)

target = ~~2~~ = 20

Worst case

Time $\propto O(N)$
Space $\propto O(1)$

arr = 3 index
arr = -1 index

Binary Search

12 August 2023 13:21

→ array should be sorted.



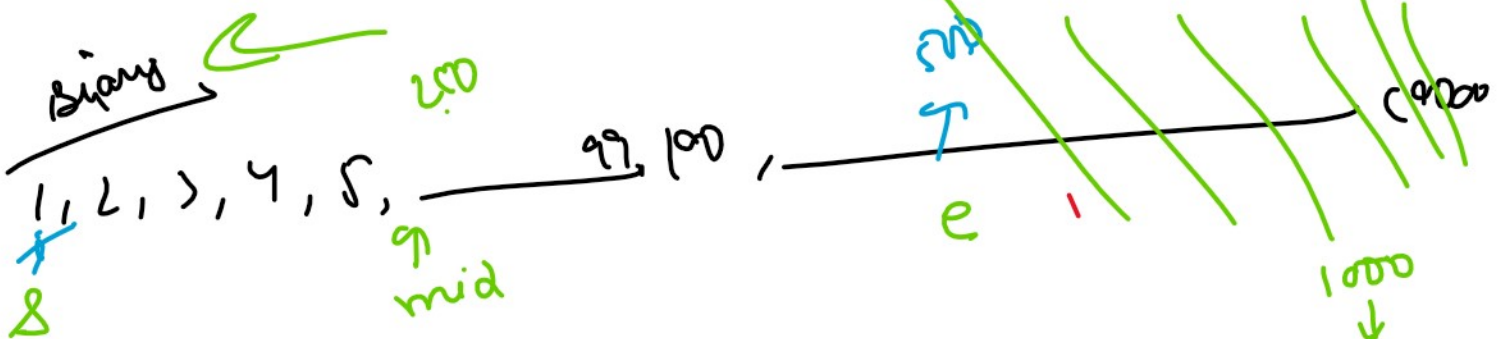
Linear Search

1, 2, 3, 4, ...

100, 200, ... 1000

o(N)

target = 99



Time $O(\log N)$
 Space $O(1)$

iterations 10-12

$O(1)$ → space U

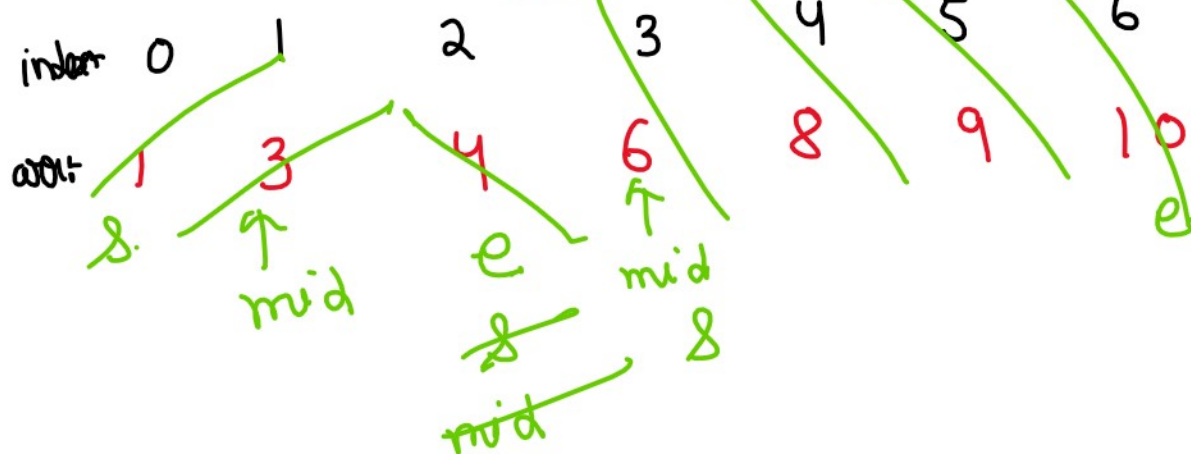


1 2 3 4 5 6 7 8 9 10
s1 s2 e1 e2

$N/2 \approx O(N)$

$N/4 \approx O(N)$

target = 5

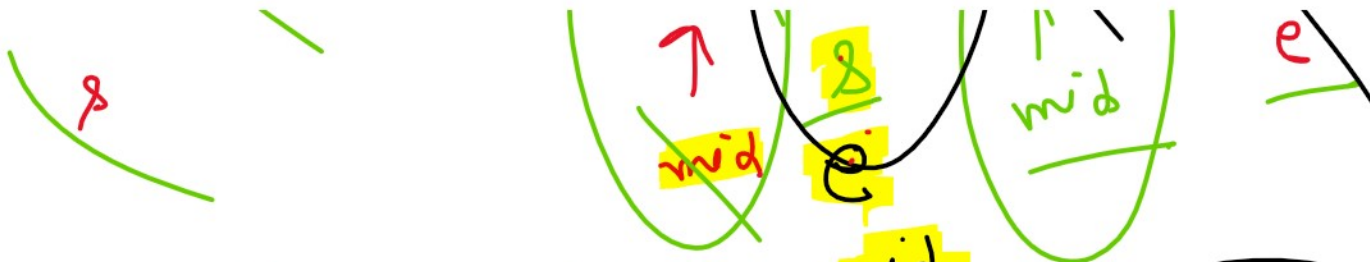


$8 > e$

while ($s \leq e$)

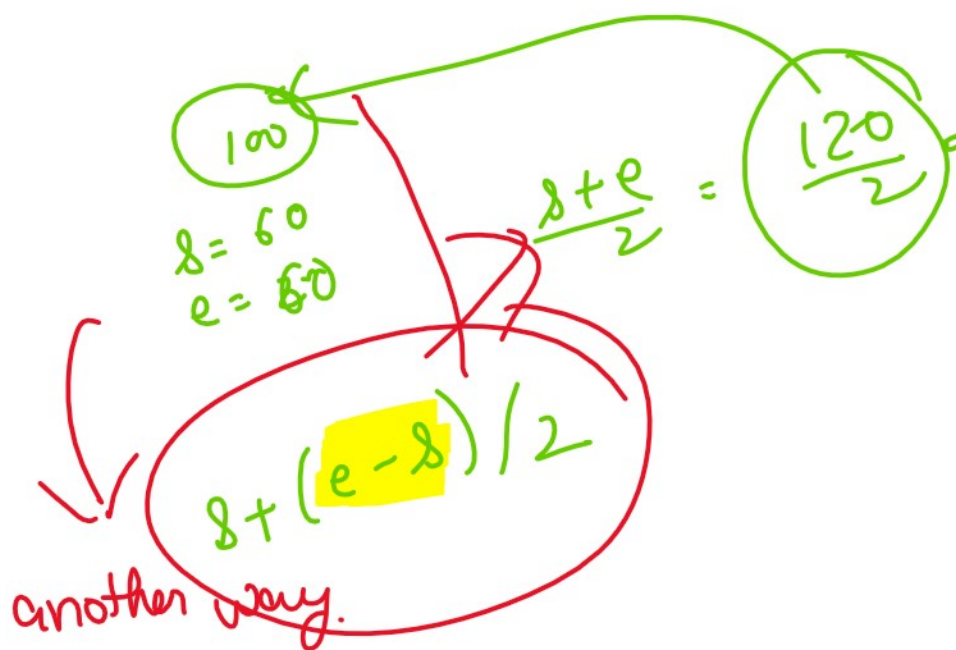
return -1 left ($e = mid - 1$)
right ($s = mid + 1$)





$$med = \frac{s+e}{2} = \frac{4+4}{2} = 4$$

4 ans



First and last occurrence of an element

12 August 2023 13:48

(in a sorted array)

index	0	1	2	3	4	5	6	7	8
arr	1	2	4	4	4	4	5	9	10
	8				mid				e

target = 4

first = 2
last = 5

first = 6
last = 6

linear Time $\rightarrow O(N)$
Space $\rightarrow O(1)$ $\rightarrow O(\log N)$
 $\rightarrow O(1)$ Binary search

\rightarrow binary search

first occurrence \rightarrow after finding the target, will go to left. $e = mid - 1$

last occurrence \rightarrow after finding the target, go to right. $s = mid + 1$

idx	0	1	2	3	4	5	6	7	8
arr	1	2	4	4	4	4	5	9	10

arr[]: 1 3 4 4 4 4 8 9 10
 s mid e mid mid
 e

first = -1/4/2
 (go left)

last = -1
 (go right)

$$\log N + \log N = 2 \log N$$

$$\downarrow$$

$$O(\log N)$$

sorted array & I asked to tell frequency of an element.

0 1 2 3 4 5 6 7
 1 3 5 6 8 8 9 10
 first last
 val = 8

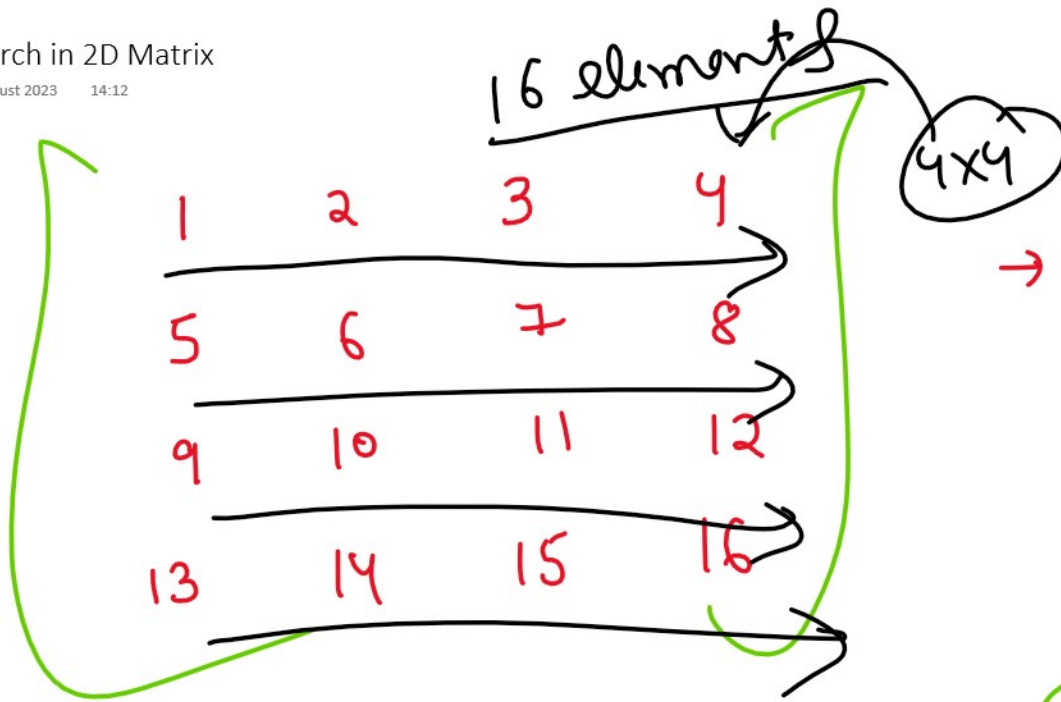
freq = 2

$$\text{last} - \text{first} + 1$$

$$5 - 4 + 1 = 2$$

Search in 2D Matrix

12 August 2023 14:12



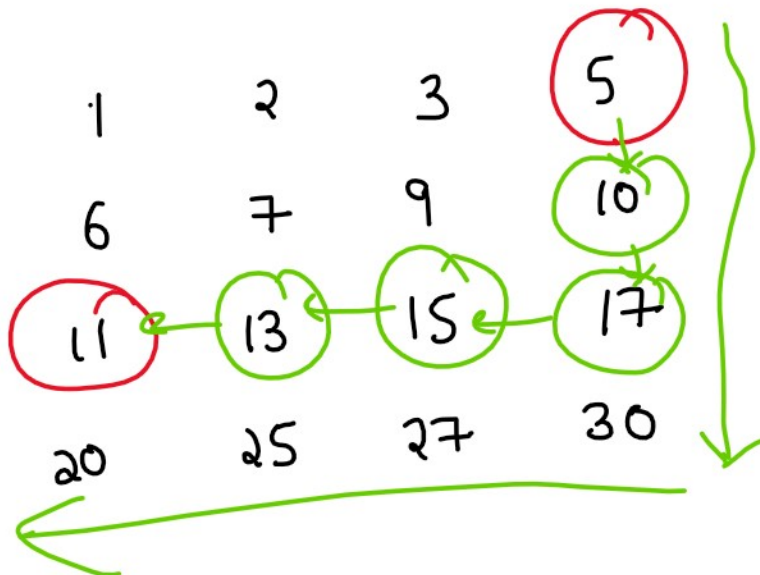
Time = $O(N \times M)$

Space = $O(1)$

$N \times \log M$

each row apply binary search.

$O(\log(N \times M))$



target = 3

$O(N \times M)$

10

greater → down

less → left

✓ ID = [1, 2, 3, 5, 27, 30]

$N \rightarrow$ rows

$M \rightarrow$ cols

$O(N \times M)$

$O(N \times M)$

✓ ID = [1, 2, 3, 5] → 2 + 1 + 0 + 0
 8
 9
 mid

$O(N \times M)$
 $O(N + M)$
 $O(\log(N \times M))$

$O(N \times M)$
 $O(\log(N \times M))$
 $O(N \times M) + O(\log(N \times M))$
 $N^3 + N^2 + N$
 Time = $O(N^3)$

0	1	2	3	5
4	6	7	9	10
8	11	13	15	17
12	20	25	27	30

target = 3

row = mid / 4
 col = mid % 4

row = 7 / 4 = 1
 col = 7 % 4 = 3

row = 3 / 4 = 0
 col = 3 % 4 = 3

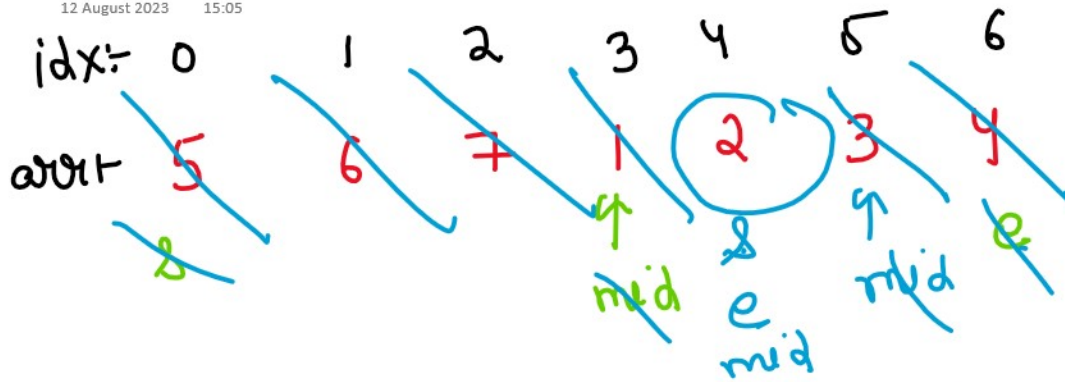
$s = 0$
 $e = 15$
 $(N \times M - 1)$
 $mid = (s + e) / 2$
 $= (0 + 15) / 2 = 7$
 $= (0 + 6) / 2 = 3$
 $= (0 + 2) / 2 = 1$
 $= 2$

$O(\log(N \times M))$
 $O(1)$

4 times

Search in rotated sorted array

12 August 2023 15:05



1 rotation point
 target = 2
 ans = 4

- check for the sorted half. (left, right)
- whether target lies in sorted half.
- Adjust (s, e)

target >= mid
 target <= end

① Sort = $O(N \log N)$

② Linear = $O(N)$

③ Binary search = $O(\log N)$

→ 1 2 3 4 5 6 7

input 5 6 7 1 2 3 4

input 3 4 5 6 7 1 2



target = 1

Time = $O(\log N)$
 Space = $O(1)$

① Sorted half

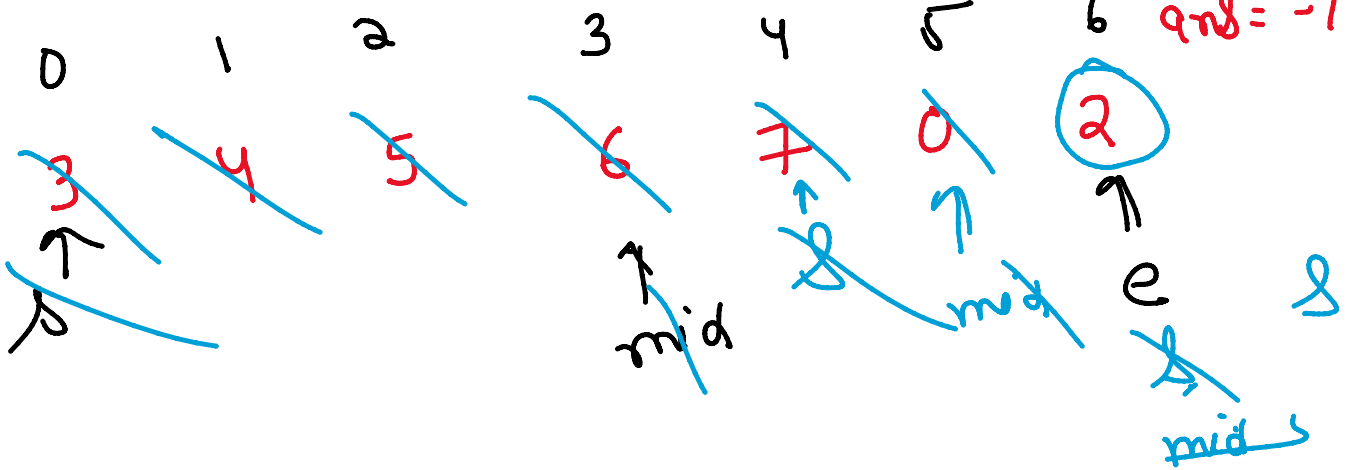
s = mid
 mid = e

ans = 1

mid <= e

target = 1

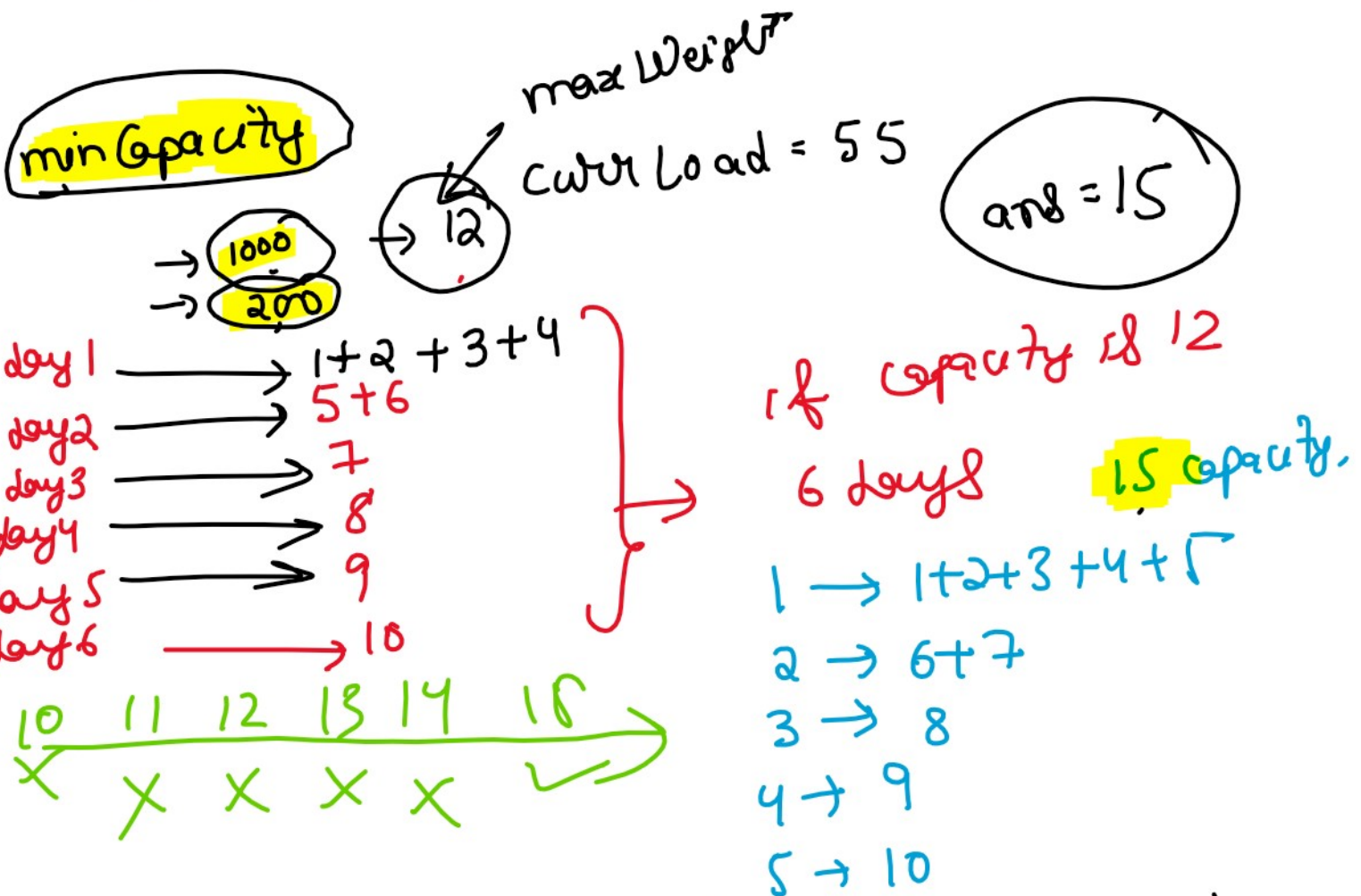
ans = -1



$s > e$
return -1

2 < 2

weights = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10], Days = 5



max weight of Array = 10

addition of All Values = 55

for (i = 10; i <= 55; i++)

(min range)

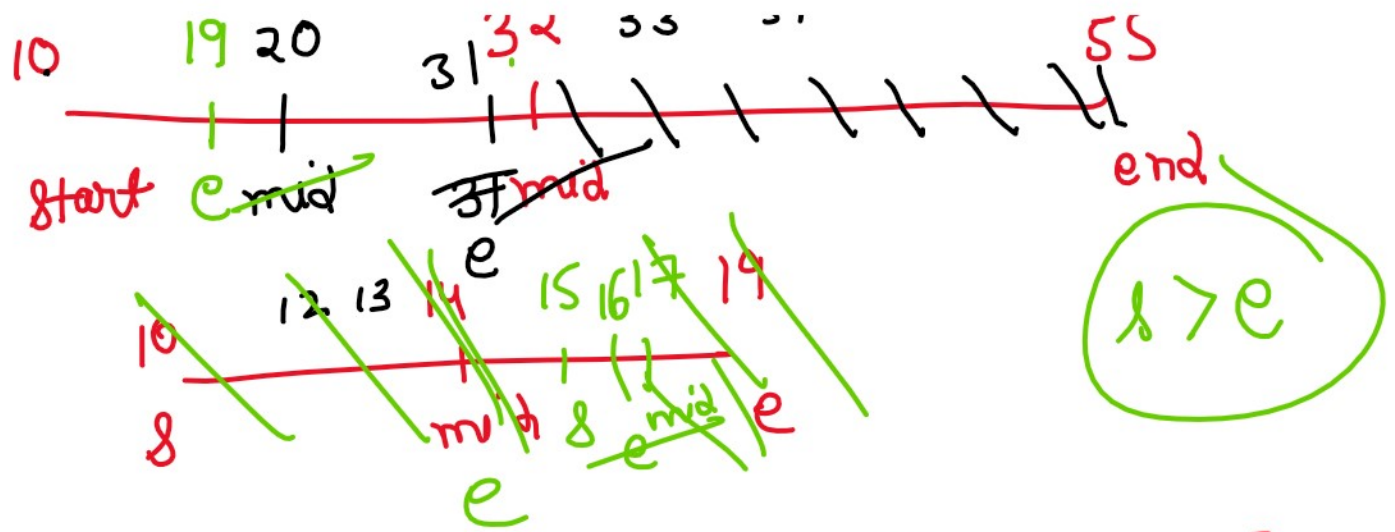
(max range)

N^2

$(10 + 55) / 2 = 32$

10 19 20 31 32 33 34 55

✓ ✓ ✓



weights = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10], Days = 5
 and = 32, 20, 17, 15

6x
 day 1 = 1 + 2 + 3 + 4
 day 2 = 5 + 6
 day 3 = 7
 day 4 = 8
 day 5 = 9
 day 6 = 10

Time = $O(N \log N)$
 Space = $O(1)$

- Aggressive Cows (Spoj) →
- Min page allocation (GFA) →
- Painter partition (GFA) →

→ GfG (microsoft interview ques
→ leetcode
→ Glass door

SDE-2

(30 to 50)

Experience
Interviews
Negotiated