

13/03 Week 5.4 Searching & Sorting

Date: 206

Class 5.1 (Searching)

i) Linear Search \rightarrow vector<int> arr;

$\rightarrow \rightarrow \rightarrow \rightarrow \rightarrow$ Found

(30 | 20 | 30 | 50 | 60 | 70 | 80)

target = 70

for ($i=0$; $i < n$; $i++$)

 if [$arr[i] == target$]
 return true;

TC $\rightarrow O(n)$

return false

ii) Binary Search \rightarrow Only Sorted Array
(may be Asc & Desc)

Left	Mid	Right
10 20 30 40 50 60 70 80	3	7
0 1 2	4 5 6	8

target = 70

(A) $e = 0^\circ$
 $e = 4^\circ$

(B) \rightarrow Mid \rightarrow $\frac{0+1}{2} = \frac{1}{2} = 0.5$

While (C) \rightarrow If Mid == Target

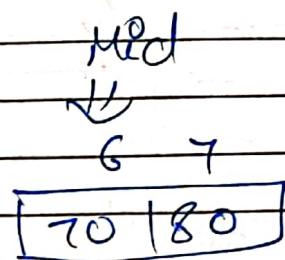
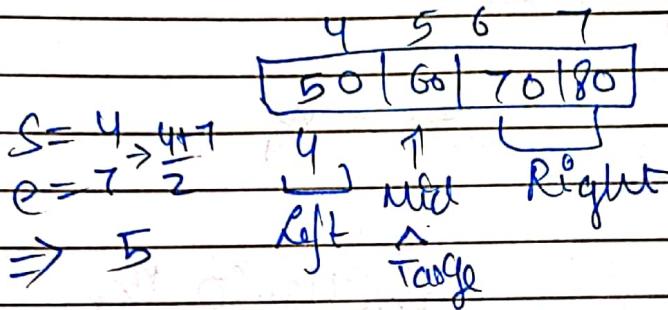
(S \leq e) return Mid / true;

13/9/23

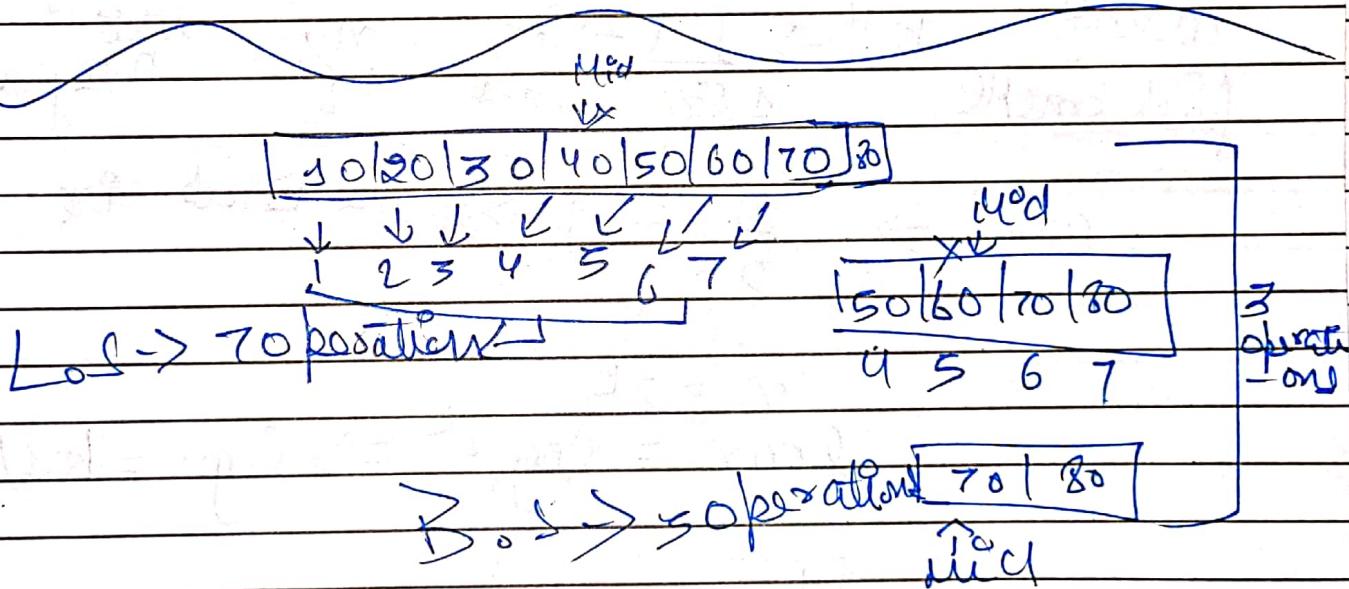
else if ($\text{Mid} < \text{Target}$)
 $s = \text{Mid} + 1;$

else ($\text{Mid} > \text{target}$)

$$e = \text{Mid} - 1$$



Target == Mid



ex) array \rightarrow 1000 size
 if target \nrightarrow present in array

LS \rightarrow 10000 comparison

BS \rightarrow 10000
 5000
 2500
 1250
 625
 312

156
 78
 39
 19
 9

4
 2
 1

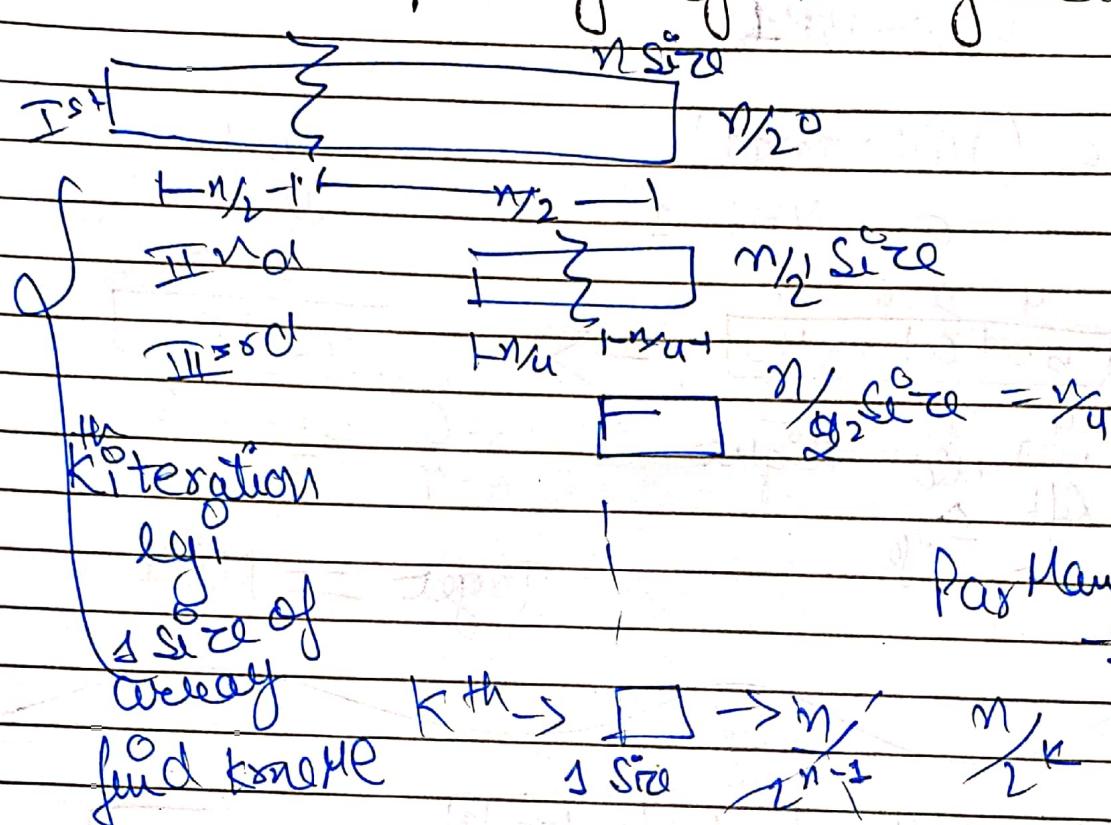
(13 comparisons)

15/9/23

APCO

Date: 2/08

Time Complexity of Binary Search



Pass Mano $\frac{n}{2^k}$
I see itna for
the Pdegaan

$$n = 2^k$$

$$\log n = \log_2 2^k \Rightarrow \log n = k(1)$$

$$k = \log n$$

$$TC \rightarrow O(\log n)$$

while ($s \leq e$)

$$\text{mid} = s + \frac{e-s}{2}$$

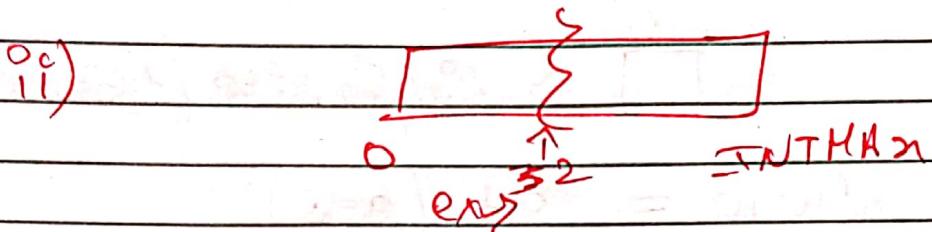
$\frac{s+e}{2} \rightarrow$ let say
we are of
size INT
max

13/9/23

+ Plan

ADCO
Date: 2024

$$i) \rightarrow \frac{s+t}{2} = \frac{0 + \text{INT_MAX}}{2} = \text{Ans}$$



Next mind $\text{int mid} = \frac{0 + \text{INT_MAX}}{2}$

overflow because int
cannot take values greater than INT_MAX

$\text{INT_MAX} + \text{Something} > \text{INT_MAX}$

So, \rightarrow we do, $\frac{\text{INT_MAX} - x}{2} \rightarrow \frac{\text{INT_MAX}}{2}$
even if $x = \text{INT_MAX}$

Let say $s \neq \text{INT_MAX}$

Then also $= \frac{\text{INT_MAX}}{2}$

So, $\left\lfloor \frac{s + (0-s)}{2} \right\rfloor > \text{INT_MAX}$
 \rightarrow always ≤ 1

$s = 32 \quad 0 = \text{INT_MAX}$

$\frac{s+t}{2} \rightarrow$ can overflow

13/9/23

APCO
Date: 210

Notes → i) $\text{start} == \text{end}$

$s \rightarrow e$



→ Single size / 1 element

ii) Mid always = $s + \frac{e-s}{2}$

should

(iii) while ($s \leq e$)

① → Find first occurrence of a No in a sorted array

Input → [10 | 20 | 30 | 30 | 30 | 30 | 30 | 40 | 50]
0 1 2 3 4 5 6 7 8

$s=0, e=7, \text{mid}=3$

$30 == 30 \rightarrow \text{True}$

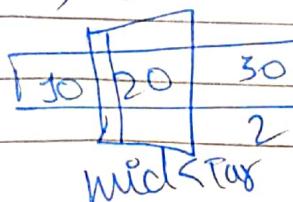
Mid Store occurrence = 3

→ go to left

Mid > Target $e = \text{mid}-1$

Mid < Target $s = \text{mid}+1$

→ $s=0, e=3-1=2, \text{mid}=1$



$s=2, e=2, \text{mid}=2$

[30] → Target = 30

13/9/23

APCO
Date: 24/11

found \rightarrow ans = 2 (new occurrence stored)

* \rightarrow we always going in left & if found element in left, the pos will change to less than past store occurrence

After that \rightarrow left mejo $e = mid - 1$
 $e = 1, s = 2$
 $e \leftarrow s \rightarrow$ Stop Loop

$$TC = O(\log n)$$

ii) Last occurrence

10 120 130 130 130 130 140 150
0 1 2 3 4 5 6 7 8
mid
 $s = 0 \rightarrow mid = 4$
 $arr(mid) == target \rightarrow found$
 $so = 130$
 $store = mid$
 $change \leftarrow e = mid + 1$

Mid > Target $\rightarrow e = mid - 1$

Mid < Target $\rightarrow s = mid + 1$

B/g/B

(S) \rightarrow Find Total occurrence

1	1	2	1	2	1	3	1	4
0	1	2	3	4	5			

$$\text{Total occ} = 3$$

$$= \text{last occ} - (\text{first} + 1)$$

$$= 3 - 1 + 1 = 3$$

(B) \rightarrow Find missing element in a sorted array

search

elements

\rightarrow arr [1 | 2 | 3 | 4 | 6 | 7 | 8 | 9] $\leftarrow 1-n$

0+1 1+1 2+1 3+1 4+1 5 6 7 \leftarrow size

($\rightarrow n$) [+1] \leftarrow Pattern break ($\rightarrow n-1$)

Approach \rightarrow i) L.C.S

ii) SUM

$\rightarrow O(n)$

given

iii) XOP

Best iv) B.S $\rightarrow O(\log n)$

\rightarrow If this sorted Array \rightarrow

\Rightarrow jab Missing element mila aya $\rightarrow i+1$

aya \rightarrow Pattern

changed

0	1	2	3	4	5	6	7
1	2	3	4	6	7	8	9

$$l = 0 \rightarrow \text{mid} = 3$$

$$r = 7 \rightarrow \text{mid} = 3$$

$$arr[\text{mid}] = \text{mid} : = 1$$

$$s = \text{mid} + 1$$

$$\text{else } \rightarrow e = \text{mid} - 1$$

$$\text{store} = \text{mid}$$

13/9/23

(213)

⇒ If we know Missing No ke Na honge
Co pattern change hoga h

4	5	6	7
6	7	8	9

$$s=4 \rightarrow 4+7 = 5 \rightarrow \text{Mid}$$

$$7-5=2 \rightarrow \text{store} = \text{Mid} = 5$$

$$\text{ans}(\text{Mid} - \text{Mid}) = 2$$

$$e = \text{Mid} - 1$$

⇒ & Also, know gis index ke honge ans + 1 kro
Tch element honge ← sorted

$$\Rightarrow s = 4 \quad e = 4 \rightarrow \text{mid} = 4 \quad s \rightarrow \boxed{6} \leftarrow e$$

$$\text{diff}(\text{arr}[\text{mid}] - \text{mid}) = 6 - 4 = 2$$

$$\text{storeAns} = \text{mid} = 4$$

$$e = \text{Mid} - 1 = 3, s = 4 \rightarrow \text{stop loop}$$

$$\Rightarrow \text{Ans} \Rightarrow \text{mid} + 1 \rightarrow \text{arr}[\text{mid}] - 1$$

If doing with for index, If doing with
element value

H/w

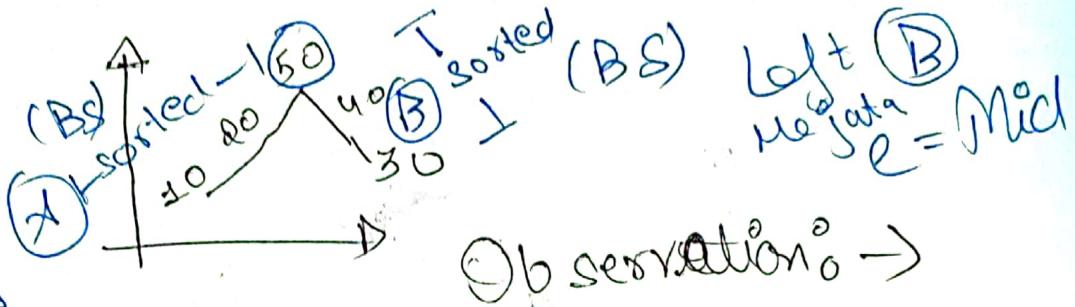
$$\boxed{\text{TC} = O(\log n)}$$

13/9/23

214

★ ★ \rightarrow Peak Element in a Mountain Array
★ ★ \rightarrow Peak Element

i/p \rightarrow [10 | 20 | 50 | 40 | 30]



(A) $\rightarrow arr[i] < arr[i+1]$

(B) $\rightarrow arr[i] > arr[i+1] \Rightarrow arr[i] > arr[i+1]$
peak point $\rightarrow arr[i-1] < arr[i] > arr[i+1]$
e = mid

$\Rightarrow arr[i] < arr[i+1] \rightarrow A$

↓
Not Case

$arr[i] > arr[i+1] \rightarrow$ can be Peak or (B)

mid
[10 | 20 | 50 | 40 | 30]
0 1 2 3 4

$e = 4 > \text{mid} = 2$

$arr[\text{mid}] < arr[\text{mid} + 1]$

$S = \text{mid} + 1 \leftarrow \text{True}$
not a Total $\leftarrow 50 < 40 \rightarrow \text{False} \rightarrow B$
or peak

14/9/23

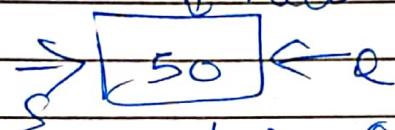
APCO
Date: 21/5

$e = \text{mid}$ \rightarrow because can be on peak

If do, $e = \text{mid} - 1$ $\Rightarrow \text{mid}$
 \downarrow
 mid

So, in $\text{mid} - 1$, I can find peak

$\leftarrow \text{mid}$



\hookrightarrow single element \rightarrow answer



Note $\rightarrow s = \text{mid}$ or $e = \text{mid}$

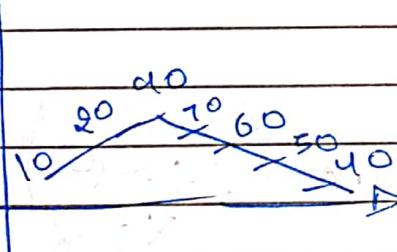
always $\Rightarrow s < e$



i) RY-RUN \rightarrow [10 20 30 40] 70 160 50 40
 \swarrow \searrow mid
 $s=0, e=7 \rightarrow \text{mid} = 3$

$\text{arr}[\text{mid}] < \text{arr}[\text{mid} + 1]$

$40 < 70 \rightarrow \text{false} \Rightarrow e = \text{mid}$



(ii) [10 20 30 40]
 $s=0, e=3, \text{mid}=1$

$\text{arr}[\text{mid}] < \text{arr}[\text{mid} + 1]$

$20 < 30 \rightarrow \text{true} \Rightarrow s = \text{mid} + 1$

(iii) [10 30 40]
 $s=0, e=3, \text{mid}=2$

$\text{arr}[\text{mid}] < \text{arr}[\text{mid} + 1]$

$30 < 40$

$s = \text{mid} + 1$

18/9/23

APCO
Date 216

$s \leftarrow \boxed{90}$ $\rightarrow e$

$s = 4, e = 4 \rightarrow \text{while } (s < e)$

out of loop

But let's say $\rightarrow \text{while } (s \leq e)$

$s \rightarrow \boxed{90} e \rightarrow$

$s = 4, e = 4, mid = 4$

$\text{arr}[mid] < \text{arr}[mid+1]$

$90 < 70 \rightarrow \text{false}$

$e = mid = 4$

$s \rightarrow \boxed{90} e$

$s = 4, e = 4, mid = 4$

$\text{arr}[mid] \leq \text{arr}[mid+1]$

$90 \leq 70 \rightarrow \text{false}$

$e = mid = 4$



loop

But if you wanna do $\text{while } (s \leq e)$

Then we \rightarrow If ($s == e$)

return mid / s / e;

};

Initial
while
loop
first cond "

13/a/23

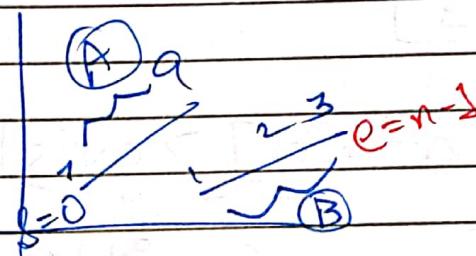
H/W → Find Pivot Element

$\begin{bmatrix} 1 & 1 & 2 & 3 & 7 & 9 \end{bmatrix} \rightarrow arr$

i/p →

$\begin{bmatrix} * & 1 & 9 & 1 & 2 & 3 \end{bmatrix}$

0 1 2 3 4



Pivot $\rightarrow 9$ or 1

koi bhi
ho skta
h

① ② ③ $arr[i] < arr[mid + 1]$ club

Pivot element $\Rightarrow arr[i] \leq arr[i+1]$

They
will
not work if (1)

If $arr[i] < arr[i+1]$

mid $\Rightarrow arr[mid] > arr[s]$
 $s = mid + 1$

L \rightarrow ① \rightarrow let say
mid \rightarrow ① mid

If Pivot = 1 $arr[mid] \leq arr[e]$
 $e = mid$

Pivot mid
mid
let say

Mid $<$ arr[e]

Pivot = 0 \Rightarrow

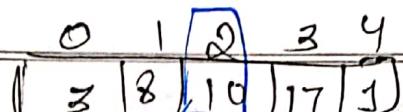
$arr[mid] \geq arr[s]$
 $s = mid$
else $\rightarrow mid - 1$

wrong
Correct $\rightarrow fg \Rightarrow 2101$

14/9/03

ARCO

Date 2/18/



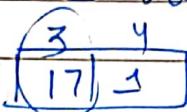
i) $s=0, e=4, \text{mid}=2$

$\text{arr}[\text{mid}] \geq \text{arr}[s]$

$10 \geq 3$

$s = \text{mid} + 1$

$s=3$
 $e=4 \rightarrow \text{mid } 3$



$\text{arr}[n] \geq \text{arr}[s]$

$s = \text{mid} + 1$

else $e = \text{mid}$

$\text{arr}[\text{mid}] \geq \text{arr}[e]$

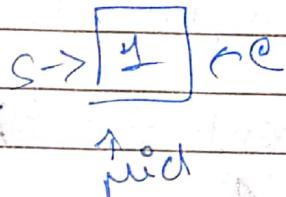
$17 \geq 14$

$s = \text{mid} + 1$

(iii)

$s=4$

$e=4 \rightarrow \text{mid } 4$



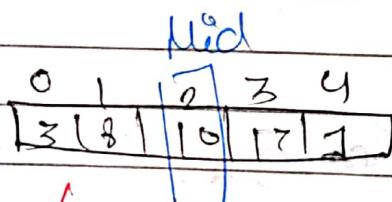
$1 \geq 1$

True

$s = \text{mid} + 1$

$s = 5, e = 4 \rightarrow \text{out of loop}$

Let's say $\rightarrow \text{arr}[\text{mid}] > \text{arr}[s]$

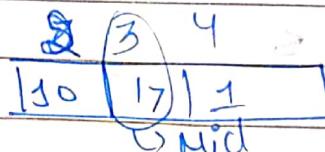


$s = \text{mid}$

else $\rightarrow e = m - 1$

Wrong example
Not working
All examples

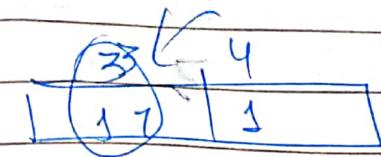
$10 > 3 \rightarrow \text{True}$
 $s = \text{mid}$



$17 > 10 \rightarrow \text{False}, s = \text{mid}$

ii)
 $s=2$

$e=4 \rightarrow 3$



(iii)

$\frac{3+4}{2} = 3$

$17 > 17 \rightarrow \text{false}$
 $e = \text{mid} - 1$

14/9/23

APCO
Date: 21/9

s e

17
mid

17 > 17 → false
 $e = m - 1$

$s = 3, e = 4 \rightarrow$ out of loop

Let's DRY Run

ex → [0 1 2 3 4]

i) $s = 0, e = 4, mid = 2$

~~1 > 7 → false~~
 $e = mid - 1$

ii)

(0) | 1 | 9 |

$s = 0, e = 1, mid = 0$

~~7 > 1 → false~~

①

2 3

Max Element Index

Let say if $arr[mid] < arr[e]$

$e = mid + 1$

1 2 3
0 = m + 1

✓ else if $arr[mid] > arr[e]$

$s = mid$

m se

Correct

else → $e = mid - 1$; if ($s == mid$)

Alternate approach for Min (Pivot) Element index

$arr[mid] < arr[e]$

$e = mid$

$(arr[mid] > arr[e])$ else → $s = mid + 1$