

Interview Questions for Binary Search:

1) Ceiling: Ans: ceiling.java

* The smallest element in array are greater than or equal to target.

Eg:- [2, 3, 8, 9, 14, 16, 18]

1. target = 14, ceiling = 14

2. target = 15, ceiling = 16.

2) Floor: Ans:- floor.java.

* The biggest number in array is smaller than or equal to target.

Eg:- [2, 3, 8, 9, 14, 16, 18]

floor target = 15, floor = 14.

3) Leetcode 744: Find Smallest Letter Greater than target.

* It is same as ceiling

Ans:- Leetcode 744.java

Sample Input: letters = ["c", "f", "j"], target = "o"

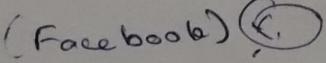
Output: "c"

+ Note that the letters wrap around.

if target == "z", but letters = ["a", "b"]

then output = "a".

1:00:00

4) Leetcode 34: Find the first and last position of element in sorted array. (Facebook) 

Ans:- Leetcode34.java

Sample Input: nums = [5, 7, 7, 8, 8, 10], target = 8

Output: [3, 5]

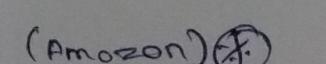
Let's take

arr = [5, 7, 7, 8, 8, 10] target = 7

first occurrence of 7

mid = 3

1:31:00

5) Greeks for Greek : find position of an element in a sorted array of infinite numbers. (Amazon) 

Ans: find Target In Infinite Array.java

Sohu:

+ Here we don't know start or end

+ then we need to find start and end.

+ There is only way to find the start and end where target will lie by taking ~~and~~ start and end with random or constant size and check.

+ But that is not efficient because it will take $O(n)$

. But we can do the reverse of binary search approach.

Binary Search decreases its size to the single element that will take $O(\log n)$. Like that we are exponentially increase the size from small to large until we find target orange.

This will also take $O(\log N)$.

+ You got start, end then apply binary search.

Eg:- $\begin{matrix} & x & & x \\ \text{arr} = & \left[\begin{matrix} 2, 3 \end{matrix} \right], \left[\begin{matrix} 5, 6, 7, 8 \end{matrix} \right], \left[\begin{matrix} 10, 11, 12, 15, 20, 23, 30 \end{matrix} \right] \end{matrix}$

apply binary search

target = 15

start = 0 if target \neq start and end

end = start + 1

~~start = end + 1~~

if target $<$ start

apply binary search

~~end = end * 2 + 1~~

\checkmark

\times end = (end - start + 1) / 2
+ end

6) Leetcode 852 : Peak Index in a Mountain Array

Ans:- Leetcode 852.java

Sample Input: arr = [0, 1, 0]

Output: 1.

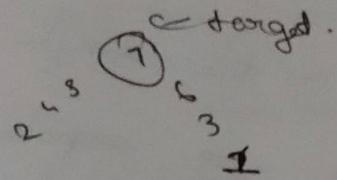
1: 58:00

+ Mountain Array is Bitonic. It increases then decreases

like mountain.

Eg:- [2, 4, 5, 7, 6, 3, 2]

* The question is to find peak index:



+ The value will not be duplicate.

Check:

① If $\text{arr}[\text{mid}] > \text{arr}[\text{mid} + 1]$

→ It is in decreasing part

so $\text{end} = \text{mid}$

② If $\text{arr}[\text{mid}] < \text{arr}[\text{mid} + 1]$

→ It is in ascending part.

Start = mid + 1

// Here I know next element is greater
but not above,

③ If $\text{arr}[\text{mid}] > \text{arr}[\text{mid} - 1] \& \text{arr}[\text{mid}] > \text{arr}[\text{mid} + 1]$

① and ② will make start, end to

point on the highest element. So you return either start or end.

7) Leetcode 1093 : Find in Mountain array:

(Hard)

Ans:- Leetcode 1093.java

Sample Input: $\text{arr} = [1, 2, 3, 4, 5, 3, 1]$, target = 3.

Output : 2 (only minimum index).

check:

① Find Peak element.

② Do binary search for ascending part. ($\text{start} = 0$, $\text{end} = \text{peak}$)

③ if not target found then do binary search for descending part. ($\text{start} = \text{peak} + 1$, $\text{end} = \text{arr.length} - 1$)

💡 This problem is not submitted in Leetcode due to different concept other than Binary Search is involving.

8) Leetcode 162 : Find Peak Element:

Ans:- Leetcode 162.java.

* The answer is same as Leetcode 852.java, because both find Peak index.

Q) Leetcode 33: Search in Rotated Sorted Array.

Ans:- Leetcode 33.java. (Amazon, Google) X

Sample Input: arr = [4, 5, 6, 7, 0, 1, 2], target = 4
Output = 4

2:30:00

* The value is not duplicated.

Not Rotated Array: [2, 4, 5, 7, 8, 9, 10, 12]

After 1 rotation in clockwise direction:

[12, 2, 4, 5, 7, 8, 9, 10]

2nd = [10, 12, 2, 4, 5, 7, 8, 9]

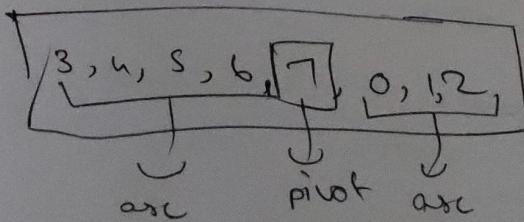
Approach:

① Find the pivot in the array



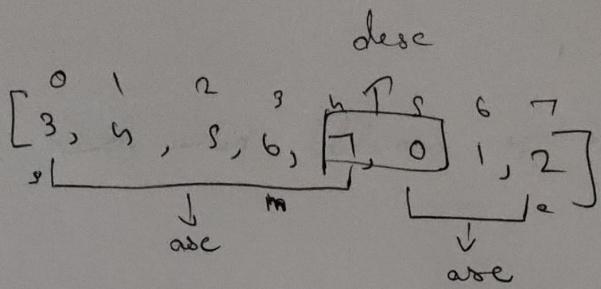
(where the element is changing).
or largest.

Here 12 is pivot.



* Search in first half with simple Binary Search (o , pivot).

* If not search second half with binary search with ($\text{pivot} + 1$, $\text{length} - 1$)



case 1: You can find ~~the~~ pivot by the method because 2 parts are ascending only the pivot and pivot + 1 will be descending. So, it is easy. and $\text{mid} < \text{end}$.

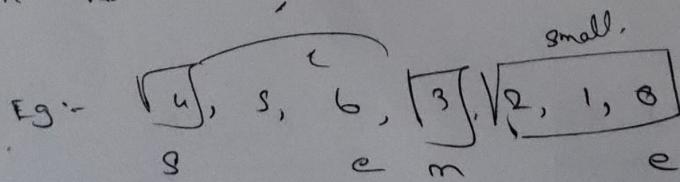
$\text{mid} > \text{mid} + 1$ return mid

case 2: if $\text{mid element} < (\text{mid} - 1)$ element. $\text{Mid} > \text{start}$.
it is also pivot.
return mid - 1.

2:43:00 :

case 3: Start element $>$ mid element.

In this case, all the element from mid will $<$ start.



Hence we can ignore all these elements we are looking for largest element.

$\text{end} = \text{mid} - 1$

case 4: start element < middle element

3, 4, 5, 6, 2
s m e

smaller than mid
ignore.

start = mid + 1.

If the mid is pivot it would have been caught in

case 1 & 2:

Hence proved, that bigger element lie ahead. Hence

ignore mid & put start = mid + 1.

3:02:00

find target using pivot cases.

case 1: pivot = target // Ans

case 2: target > start

search = Start - ~~mid~~ - 1

case 3: target < start

search = pivot + 1, end

3:08:00

Ques: Find the Rotation count in Rotated Sorted Array.

Ans: - Rotated Count.java

Input: arr [] = { 18, 18, 2, 3, 6, 12 }

Output: 2

Explanation / Approach:

18, 18, 2, 3, 6, 12
s p e

Here the pivot + 1 is the answer.

11) Leetcode 410: Split Array Largest Sum

Ans: - Leetcode 410 .java

Input: - [7, 2, 3, 10, 8], m = 2

Google

Output: - 18

Approach:

continuous,

$\frac{7, 2, 3, 10}{2n}$

$\frac{8}{8}$

largest

2n

$\frac{7, 2, 3}{m}$

$\frac{10, 8}{18}$

(18)

minimum
in largest.

7, 2

$\frac{5, 10, 8}{23}$

23

Ans

* min. no. of partitions that we can make = 1

* what is max no. of partitions that we can make = N

ans δ = [3, 4, 1, 2] \Rightarrow {3}, {4}, {1}, {2}

case 1: max value of ans of question = sum of all values in array

ans 2: min value of ans of question
= max. value in array,