

Searching

Index

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

Search = 5 -

for loop $i=0$ to length of array;
if $arr[i] == target$:
return i

return -1

Linear Search = Time Complexity = $O(n)$
Binary Search = Time Complexity = $O(\log n)$

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

start \rightarrow middle \rightarrow end

$\frac{start + end}{2} = middle$

target = 6
index = 5 -

$M \leq target \leq end$
 $start = middle + 1$

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

arr[middle] == target
eliminate

$$\frac{4+7}{2} = \frac{11}{2} = 5.5 \approx \underline{\underline{5}}$$

0	1	2	3	4	5
1	2	3	4	5	6

S \rightarrow E \rightarrow M

$$\frac{0+5}{2} = \frac{5}{2} = 2.5 \approx \underline{\underline{2}}$$

end = middle - 1

P r i . .

Time Complexity

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

$$\text{Middle} = n/2$$



$$\text{start} = 0$$

$$\text{end} = \text{length of arr} - 1$$

while $\text{start} \leq \text{end}$:

$$\text{middle} = \frac{\text{start} + \text{end}}{2}$$

if $\text{arr}[\text{middle}] == \text{target}$:

return middle

else if $\text{arr}[\text{middle}] < \text{target}$:

$$\text{start} = \text{middle} + 1$$

else if $\text{arr}[\text{middle}] > \text{target}$:

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

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

$$\frac{0+8}{2} = 4$$

Search in Rotated Sorted Array

6	7	8	1	2	3	4	5
---	---	---	---	---	---	---	---

Pivot Element $= \text{Mid} / 2$

Ap: start, end

$$6 \leq 1 \times$$

while $s \leq \text{end}$:

$$\text{mid} = (\text{start} + \text{end}) / 2$$

$mid = (start) / 2$

if $arr[mid] == target$

return mid

else if $arr[start] <= arr[mid]$:

↳ if $(arr[start] <= target)$ and $(target <= arr[mid])$:

end = $mid - 1$

else $start = mid + 1$

else if $arr[mid] < arr[end]$:

↳ if $(arr[end] >= target)$ and $(target >= arr[mid])$:

start = $mid + 1$

else
end = $mid - 1$

return -1