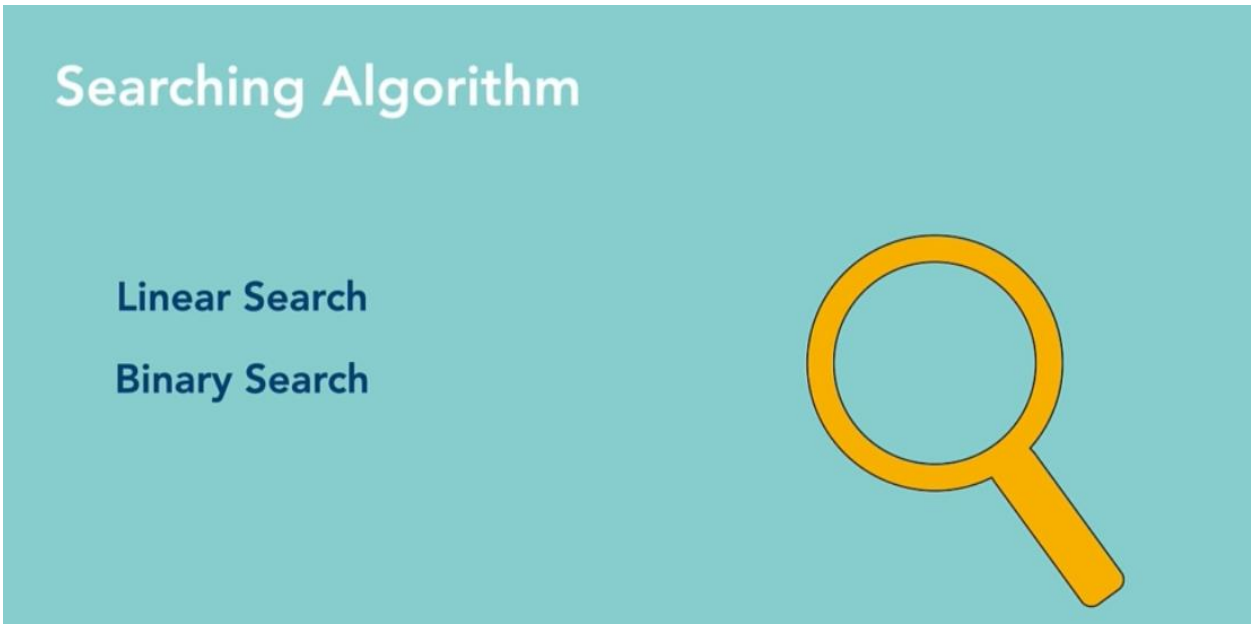
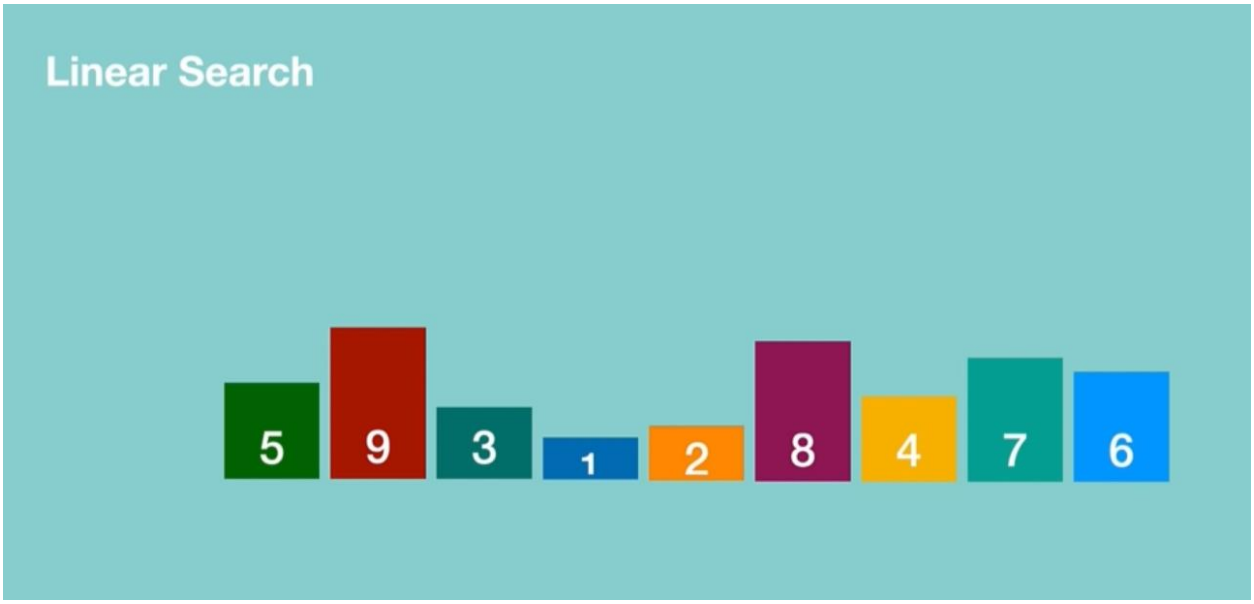




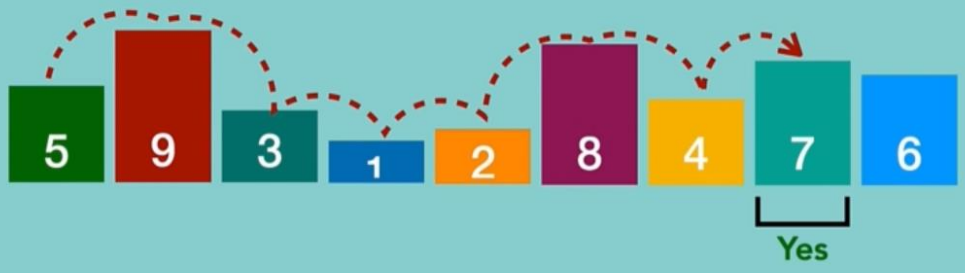
Linear search is also called as Sequential Search. This suits for both sorted and unsorted arrays.

Linear Search

Search for 7

Linear Search

Search for 7



Time complexity : O(N)
Space complexity : O(1)

Linear Search Pseudocode

- Create function with two parameters which are an array and a value
- Loop through the array and check if the current array element is equal to the value
- If it is return the index at which the element is found
- If the value is never found return -1

Time complexity: O(N)

Space complexity: O(1)

Binary Search

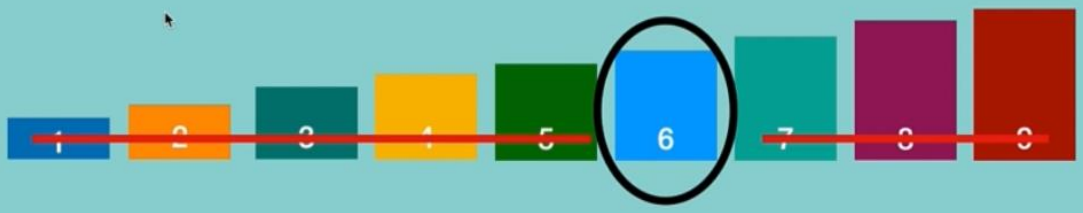
- Binary Search is faster than Linear Search
- Half of the remaining elements can be eliminated at a time, instead of eliminating them one by one
- Binary Search only works for **sorted arrays.**

Binary Search

Searching for **6**

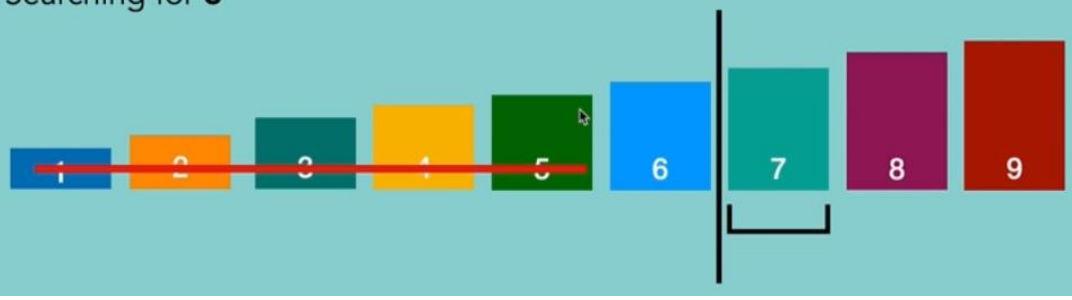
Binary Search

Searching for **6**



Binary Search

Searching for **6**



Binary Search Pseudocode

- Create function with two parameters which are a sorted array and a value
- Create two pointers : a left pointer at the start of the array and a right pointer at the end of the array.
- Based on left and right pointers calculate middle pointer
- While middle is not equal to the value and start<=end loop:
  - if the middle is greater than the value move the right pointer down
  - if the middle is less than the value move the left pointer up
- If the value is never found return -1

Binary Search Time Complexity

Worst and Average Case

O(log n)

Best Case

O(1)

Binary Search Time Complexity

Searching for 12

[2,4,5,9,11,13,14,15,19,20,22,23,27,30,32,39]

Step 1 [2,4,5,9,11,13,14,**15**,19,20,22,23,27,30,32,39]

Step 2 [2,4,5,**9**,11,13,14]

Step 3 [11,**13**,14]

Step 4 [11]

Not Found
16 elements = 4 Steps

Binary Search Time Complexity

Searching for 72

[2,4,5,9,11,13,14,15,19,20,22,23,27,30,32,39,42,44,45,49,51,53,54,55,59,60,62,63,67,70,72,79]

Step 1 [2,4,5,9,11,13,14,15,19,20,22,23,27,30,32,**39**,42,44,45,49,51,53,54,55,59,60,62,63,67,70,72,79]

Step 2 [42,44,45,49,51,53,54,**55**,59,60,62,63,67,70,72,79]

Step 3 [59,60,62,**63**,67,70,72,79]

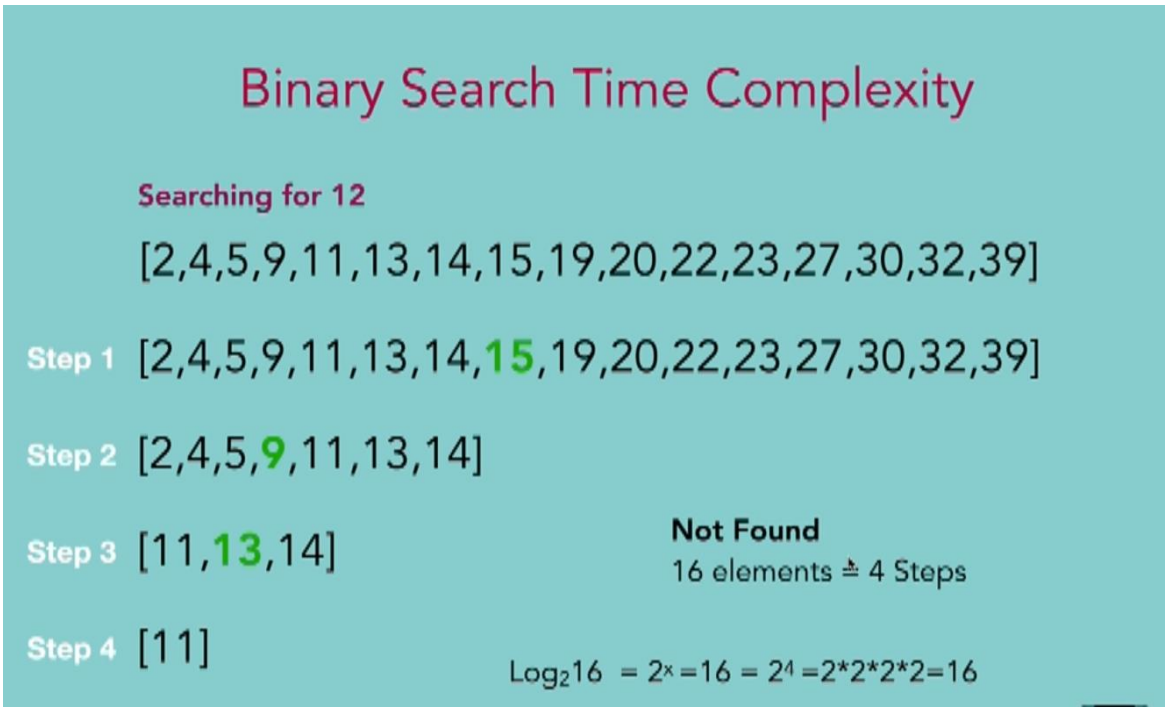
Step 4 [67,**70**,72,79]

Step 5 [**72**,79]

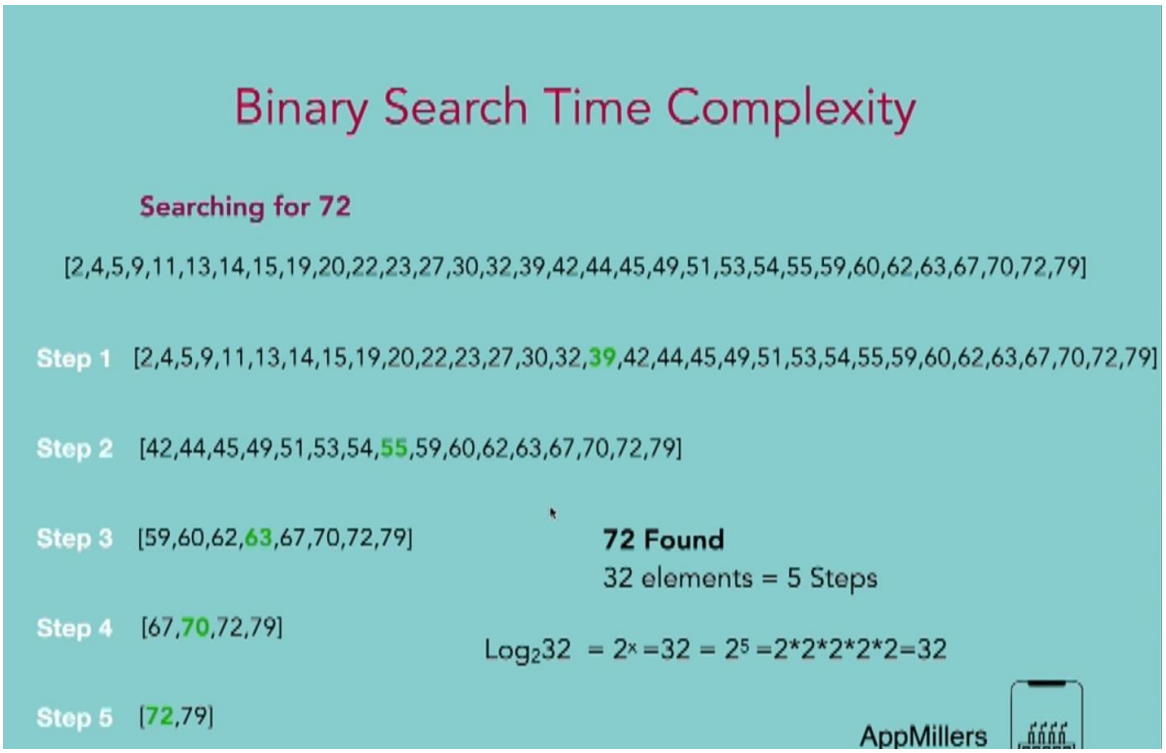
72 Found
32 elements = 5 Steps

Log_2N $2^x = N$

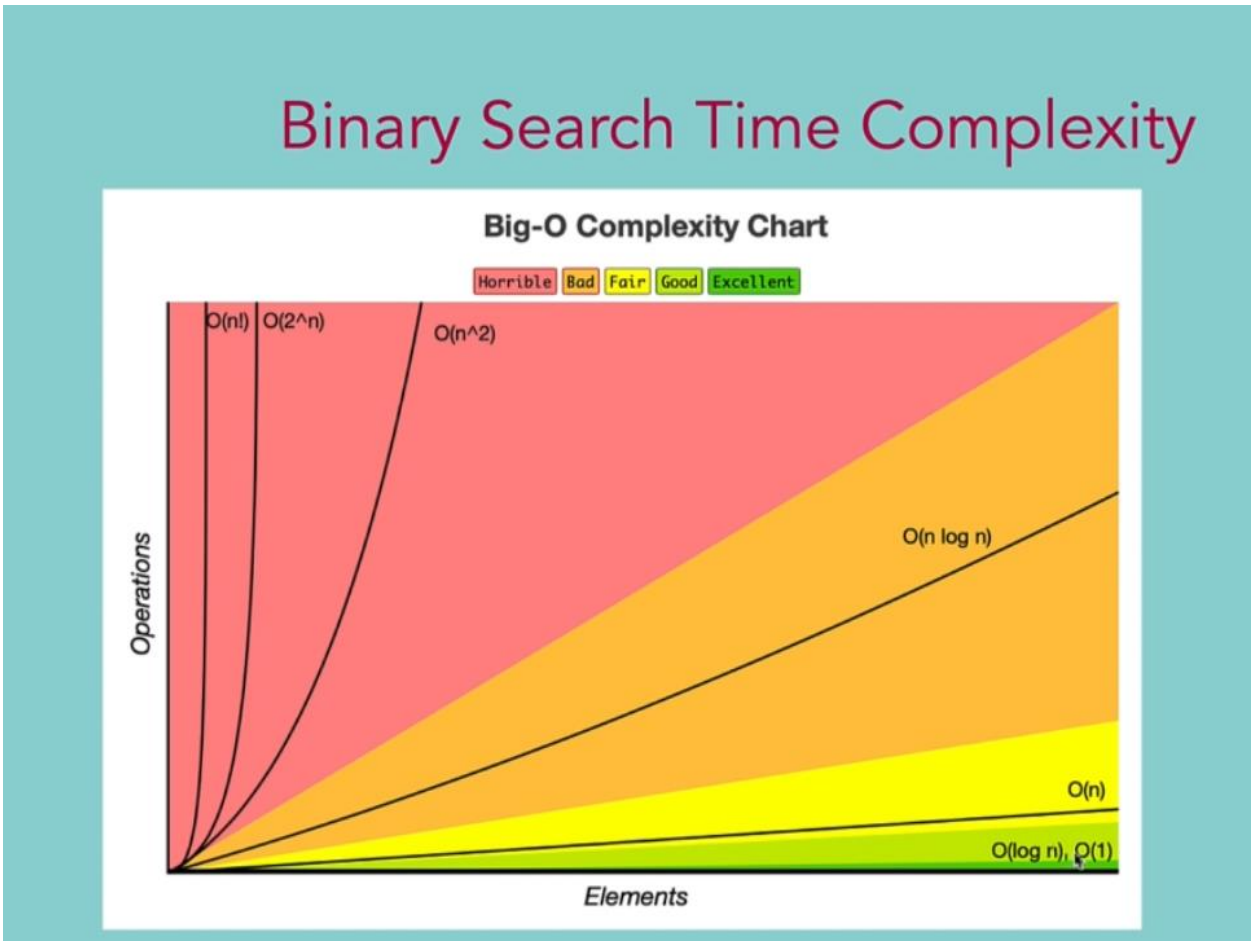
AppMillers



Here, the size of the array is 16. The time complexity for Binary Search is $\log_2$(size of the array). $\text{Log}_2(16) = 4$ → so to find the search element we need 4 steps.



Here, the size of the array is 32. The time complexity for Binary Search is $\log_2$(size of the array). $\text{Log}_2(32) = 5$ → so to find the search element we need 5 steps.



If we have a sorted array then the best searching algorithm is binary search.