Theory: Binary search

8 minutes 6 / 6 problems solved

Start practicing

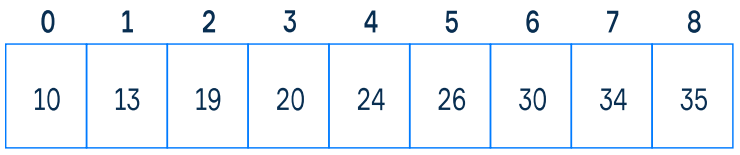
§1. Introduction

Binary search is a fast algorithm for finding an element in a **sorted array**. The algorithm runs in logarithmic time, making $O(\log n)$ comparisons, where n is the length of the input array.

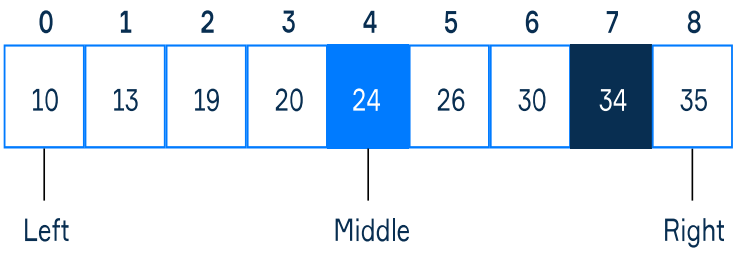
The algorithm begins by comparing the middle array element with the target value. If there is a match, it returns the element index. Otherwise, the search proceeds to the left or right subarray, depending on whether the target value is less or greater than the middle element. It goes on until it finds the target value or a new search interval is empty.

§2. Example

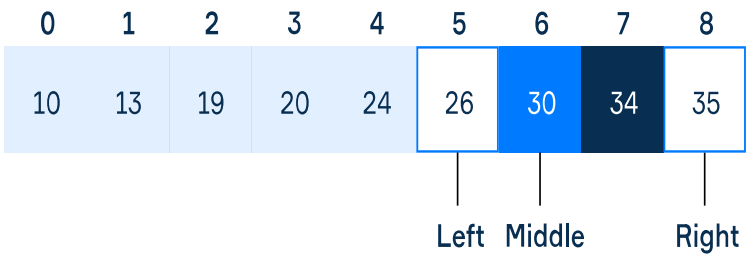
Suppose we have an integer array sorted in ascending order. We want to find the index of the value 34 with the binary search. The input array has nine elements with indices from 0 to 8. The target value 34 has index 7.



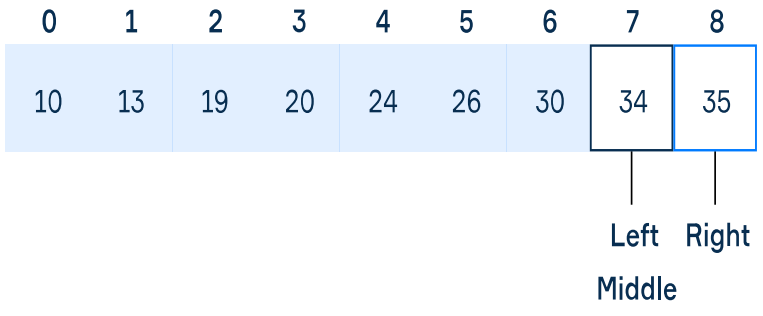
- First, we consider the entire array. The leftmost index is 0, the rightmost one is 8. The index of the middle element is $\frac{0+8}{2} = 4$



- It's time to make some decisions. Our target element 34 is greater than the middle element 24. Because the array is sorted in *ascending* order, the left subarray cannot possibly contain the target element, so we continue the search in the right subarray.
- We consider the elements with indices from 5 to 8. The index of the middle element is $\frac{5+8}{2} = 6$ (integer division).



- The target element 34 is greater than the middle element 30. Because the array is sorted in *ascending* order, we continue the search in the right subarray.
- This time we look at elements with indices from 7 to 8. The index of the middle element is $\frac{7+8}{2} = 7$ (integer division).



Look what happened: the target element 34 matches the middle value 34! Hence we return the index 7.

§3. Conclusion

Current topic:

✓ Binary search

Topic depends on:

✓ Linear search

Topic is required for:

Binary search in Java

Binary search in Python

Table of contents:

The binary search algorithm divides the array into two subarrays at each step and then searches for the element in one of them. The number of comparisons is much less than the length of the array.

If you would like to see a visualization of the algorithm, here is a visualization; input a target value and click "Binary Search"!

Report a typo

170 users liked this theory. **4** didn't like it. **What about you?**

Start practicing

Comments (4) Hints (0) Useful links (0) Show discussion