

Binary Search

Do Now

I would like to find the index where number 11 is stored in this array

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----

How would you do it?



Do Now

Let's imagine this game, the computer selects an integer value between 1 and 16 and our goal is to guess this number with a minimum number of questions. For each guessed number the computer states whether the guessed number is equal to, bigger or smaller than the number to be guessed.



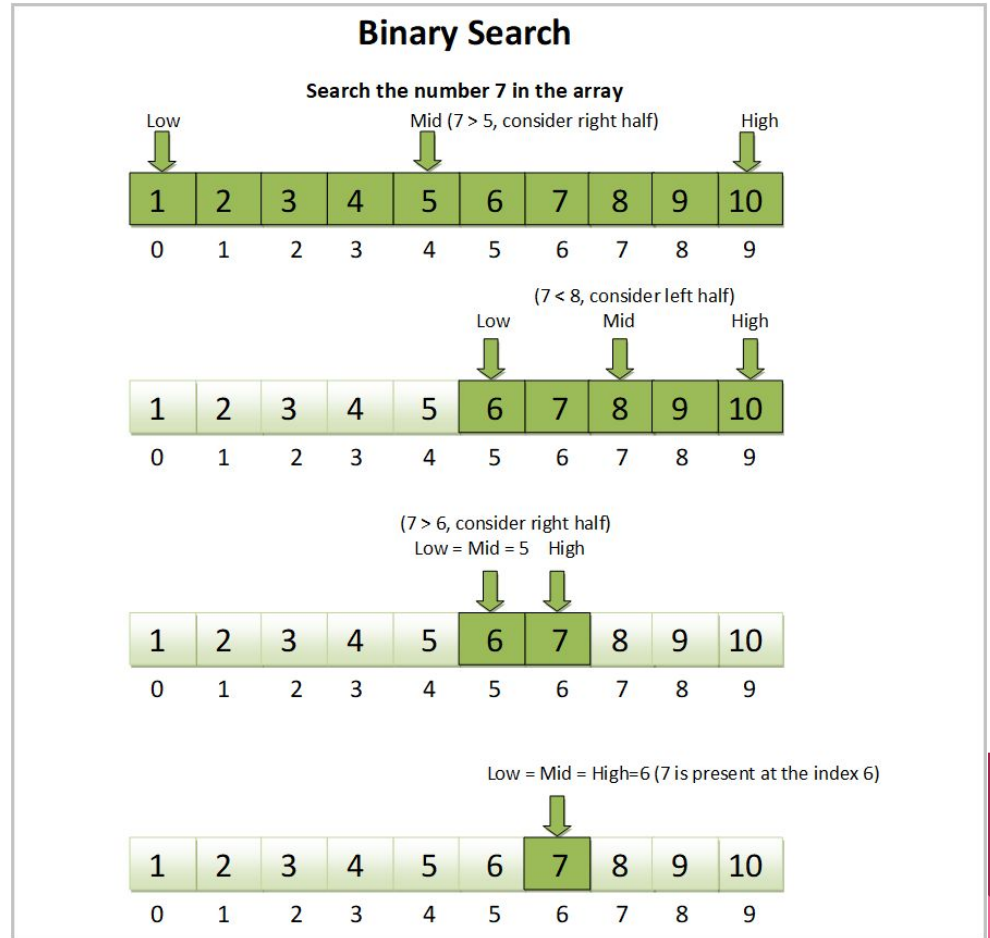
How would you do it?



Binary Search

It is an algorithm used to find a target value within a sorted list.

It repeatedly divides the search interval in half.



What values do we need implement the algorithm?

We need two variables:

- Pointer to the beginning of the list
- Pointer to the end of the collection



Binary Search - Algorithm

1. Set the **low** pointer to the beginning of the array (index 0).
2. Set the **high** pointer to the end of the array (`array.length - 1`).
3. While the **low** is less than or equal to the **end** pointer, repeat these steps:
 - a. Calculate the middle element index: **$\text{mid} = \text{low} + (\text{high} - \text{low}) / 2$** .
 - b. Compare the value at middle index (**mid**) with the target value.
 - If **`arr[mid]`** is equal to the target value, return **mid** (search successful).
 - If **`arr[mid]`** is less than the target value, set the **low** to **`mid + 1`**.
 - If **`arr[mid]`** is greater than the target value, set the **high** to **`mid - 1`**.
4. If the **low** pointer becomes greater than the **high** pointer, the target value is not in the collection. Return **-1** to indicate that the target is not present.



Time Complexity

Average Case: $O(\log n)$

Worst Case: $O(\log n)$

Best Case: $O(1)$

