

BINARY SEARCH ALGORITHM

It's a search algorithm that finds the position of a target value within a sorted array. Binary search compares the target value to the middle element of the array.

If they are not equal, the half in which the target cannot lie is eliminated and the search continues on the remaining half, again taking the middle element to compare to the target value, and repeating this until the target value is found. If the search ends with the remaining half being empty, the target is not in the array.

Binary search working

Consider the value to be searched as x

>> Compare x with the middle element.

>> If x matches with the mid element, we return the mid index.

>> Else if x is greater than the middle element.

>> Then x can only lie in right half of the sub array, after the mid element.

>> So we recur for right half.

>> Else (x is smaller) recur for the left half.

ALGORITHM

1. Take input : array , left , right & x
2. Start Loop – while (left <= right)
 - 2.1 mid = left (right – left)/2
 - 2.2 if (arr [mid] == x) then
 - 2.2.1 return mid
 - 2.3 else if (arr [mid] < x) then
 - 2.3.1 left = mid + 1
 - 2.4 else
 - 2.4.1 right = mid – 1
- End Loop
3. return -1

1	3	12	14	23	34	55	65	75	78
0	1	2	3	4	5	6	7	8	9

Step 1 – Take inputs

array = arr | left = 0 | right = 9 | x = 75

Step 2 – Loop while (left <= right)

mid = left + (right – left)/2