Binary Search

If your dataset is ordered, you can search the "middle" element (halfway through the dataset) and check if it is "greater" or "lesser" than your search value. Continue cutting your dataset in half until the "middle" element is either the search value, or nothing remains to search.

NOTE:
Since using integer division when calculating index midpoint, you must always include the midpoint when splitting the lower half

Time Complexity

$O(\log N)$

$L, R = 0, N$
$M = (L + (R-L))//2$

Example

[1, 2, 3, 4, 5, 6, 7]
Search for 5 (R)

middle = 4, 4 < 5 so search [5, 6, 7]
middle = 5, 6 > 5 so search [5]
middle = 5, 5 == 5 so return True!

[1, 2, 3, 4, 5, 6, 7, 8]
search for 5
[5, 6, 7, 8]
[5, 6]