

Binary Search \rightarrow Recursion

Sorted Array

Reduce
search
space

0 1 2 3 4
10 20 40 50 70

$i = 0$

$j = 4$

\uparrow
mid

$$0 + 4 // 2 = 2$$

Divide & Conquer

\hookrightarrow 2 halves

$O(1)$

$BS(arr, i, j, key):$

$$mid = i + (j - i) // 2$$

$key == arr[mid]:$

return mid

$O(1)$

$T(n/2)$

$key < arr[mid]:$

Left subarray
OR

$BS(arr, i, mid - 1, key)$

Recursion

$T(n/2)$

$key > arr[mid]:$

Right subarray

$BS(arr, mid + 1, j, key)$

key = 50
0
 \hookrightarrow 3

key = 50 \leftarrow small Problem

1

2

3

$$T(n) = T\left(\frac{n}{2}\right) + c$$

$$\left. \begin{array}{c} \uparrow \\ \downarrow \end{array} \right\} k = \log_2 n$$

$$= T(1) + c \log_2 n$$

$$= \underline{O(\log_2 n)}$$