Binary Search : Pre-requisite : Sorted Array : Follows divide and conquer technique

01 April 2024   20:09

Key = 77

	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]
	11	22	33	44	55	66	77	88	99

left .... Left sub array .... mid .... Right sub array .... right

1) Get the key from the user
2) Calculate left index, right index and mid index
   a. Left = 0
   b. Right = 8
   c. Mid = (left + right) /2
      i. = (0+8)/2 = 4
         Mid = 4
3) Compare the key with the element at mid
   a. 77 == arr[mid]
      <mark>77 == 55 ?</mark>
      If yes, return the index
      If no, got to step 4

4) Check if the key is smaller to mid element or greater
   a. If the key is smaller to mid element, continue the search in left subarray
   b. If the key is greater to the mid element, continue the search in the right sub array

[5]	[6]	[7]	[8]
66	77	88	99

Left   mid   right

As the key is Greater, consider the right sub array
Right sub array starts from mid+1 to right index

Start from step 2:
Left = mid+1 = 5
Right = 8
Mid = (left + right)/2
   = (5+8)/2
   = 6

3) Compare the key with the element at mid
   77 == arr[mid]
   <mark>77 == 77</mark>

   I found the element at index 6