Prompt   Scratchpad   Our Solution(s)   Video Explanation   Run Code

Solution 1   Solution 2

```go
1  // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3  package main
4
5  // O(log(n)) time | O(log(n)) space
6  func ShiftedBinarySearch(array []int, target int) int {
7      return helper(array, target, 0, len(array)-1)
8  }
9
10 func helper(array []int, target int, left int, right int) int {
11     if left > right {
12         return -1
13     }
14     middle := (left + right) / 2
15     potentialMatch := array[middle]
16     leftNum, rightNum := array[left], array[right]
17     newleft, newright := middle+1, right
18     if target == potentialMatch {
19         return middle
20     } else if leftNum <= potentialMatch {
21         if target < potentialMatch && target >= leftNum {
22             newleft, newright = left, middle-1
23         }
24     } else {
25         if !(target > potentialMatch && target <= rightNum) {
26             newleft, newright = left, middle-1
27         }
28     }
29     return helper(array, target, newleft, newright)
30 }
31
```

```go
1  // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3  package main
4
5  // O(log(n)) time | O(log(n)) space
6  func ShiftedBinarySearch(array []int, target int) int {
7      return helper(array, target, 0, len(array)-1)
8  }
9
10 func helper(array []int, target int, left int, right int) int {
11     if left > right {
12         return -1
13     }
14     middle := (left + right) / 2
15     potentialMatch := array[middle]
16     leftNum, rightNum := array[left], array[right]
```