

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1

Solution 2

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 // O(log(n)) time | O(log(n)) space
4 function shiftedBinarySearch(array, target) {
5   return shiftedBinarySearchHelper(array, target, 0, array.length - 1
6 }
7
8 function shiftedBinarySearchHelper(array, target, left, right) {
9   if (left > right) return -1;
10  const middle = Math.floor((left + right) / 2);
11  const potentialMatch = array[middle];
12  const leftNum = array[left];
13  const rightNum = array[right];
14  if (target === potentialMatch) {
15    return middle;
16  } else if (leftNum <= potentialMatch) {
17    if (target < potentialMatch && target >= leftNum) {
18      return shiftedBinarySearchHelper(array, target, left, middle - 1
19    } else {
20      return shiftedBinarySearchHelper(array, target, middle + 1, righ
21    }
22  } else {
23    if (target > potentialMatch && target <= rightNum) {
24      return shiftedBinarySearchHelper(array, target, middle + 1, righ
25    } else {
26      return shiftedBinarySearchHelper(array, target, left, middle - 1
27    }
28  }
29 }
30
31 exports.shiftedBinarySearch = shiftedBinarySearch;
32
```

Solution 1

Solution 2

Solution 3

```
1 function shiftedBinarySearch(array, target) {
2   // Write your code here.
3 }
4
5 // Do not edit the line below.
6 exports.shiftedBinarySearch = shiftedBinarySearch;
7
```

Our Tests

Custom Output

Submit Code

1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.

2

3 // O(log(n)) time | O(log(n)) space

4 function shiftedBinarySearch(array, target) {

5 return shiftedBinarySearchHelper(array, target, 0, array.length - 1

6 }

7

8 function shiftedBinarySearchHelper(array, target, left, right) {

9 if (left > right) return -1;

10 const middle = Math.floor((left + right) / 2);

11 const potentialMatch = array[middle];

12 const leftNum = array[left];

13 const rightNum = array[right];

14 if (target === potentialMatch) {

15 return middle;

16 } else if (leftNum <= potentialMatch) {

17 if (target < potentialMatch && target >= leftNum) {

18 return shiftedBinarySearchHelper(array, target, left, middle - 1

19 } else {

20 return shiftedBinarySearchHelper(array, target, middle + 1, righ

21 }

22 } else {

23 if (target > potentialMatch && target <= rightNum) {

24 return shiftedBinarySearchHelper(array, target, middle + 1, righ

25 } else {

26 return shiftedBinarySearchHelper(array, target, left, middle - 1

27 }

28 }

29 }

30

31 exports.shiftedBinarySearch = shiftedBinarySearch;

32

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

