# Search in Rotated Sorted Array II

Try to solve the Search in Rotated Sorted Array II problem.

# We'll cover the following ^ Statement Examples Understand the problem Figure it out! Try it yourself

#### **Statement**

You are required to find an integer t in an array arr of non-distinct integers. *Prior* to being passed as input to your search function, arr has been processed as follows:

- It has been sorted in non-descending order.
- It has been rotated around some pivot k, such that, after rotation, it looks like this: [nums[k], nums[k+1], ..., nums[n-1], nums[0], nums[1], ..., nums[k-1]]. For example, [10, 30, 40, 42, 42, 47, 78, 90, 901], rotated around pivot k = 5 becomes [47, 78, 90, 901, 10, 30, 40, 42, 42].

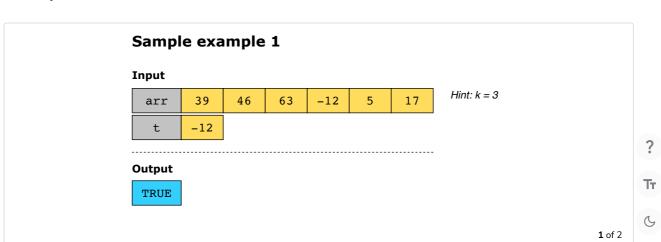
Return TRUE if t exists in the rotated, sorted array arr, and FALSE otherwise, while minimizing the number of operations in the search.

**Note:** In this problem, the value of k is not passed to your search function.

#### Constraints

- $1 \leq \operatorname{arr.length} \leq 5000$
- $-10^4 \le arr[i] \le 10^4$
- arr is guaranteed to be rotated at some pivot index.
- $-10^4 \le t \le 10^4$

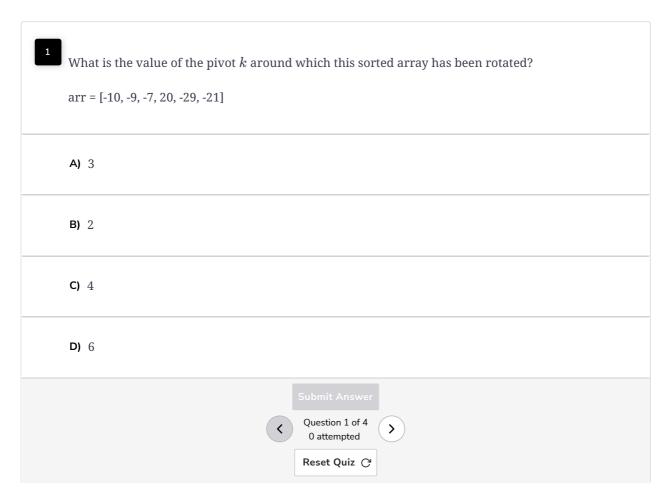
### **Examples**





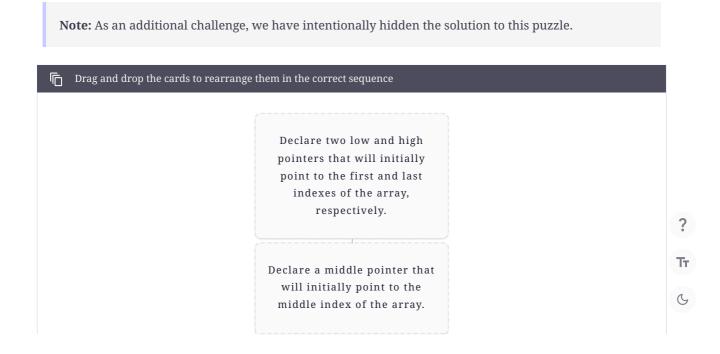
## Understand the problem

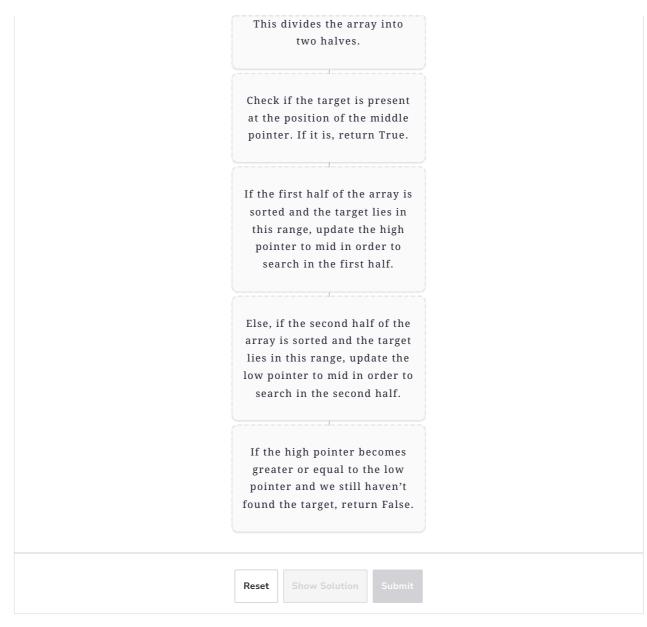
Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:



# Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.





# Try it yourself

Implement your solution in the following coding playground.

**Note:** We have left the solution to this challenge as an exercise for you. You may try to translate the logic of the solved puzzle into a coded solution.



