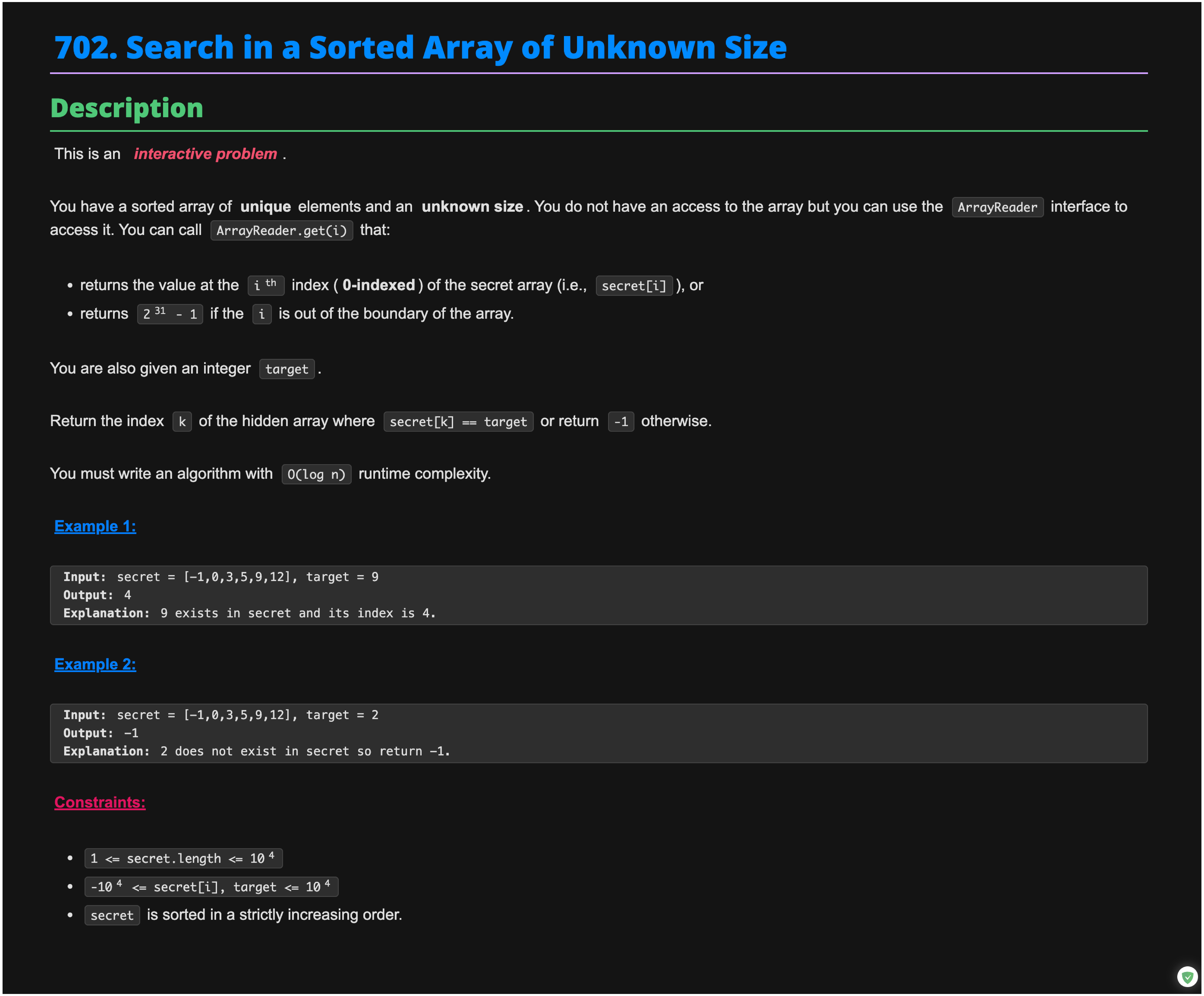

702. Search in a Sorted Array of Unknown Size

Description

This is an *interactive problem* .

You have a sorted array of **unique** elements and an **unknown size** . You do not have an access to the array but you can use the `ArrayReader` interface to access it. You can call `ArrayReader.get(i)` that:

- returns the value at the i^{th} index (**0-indexed**) of the secret array (i.e., `secret[i]`), or
- returns $2^{31} - 1$ if the `i` is out of the boundary of the array.

You are also given an integer `target` .

Return the index `k` of the hidden array where `secret[k] == target` or return `-1` otherwise.

You must write an algorithm with `O(log n)` runtime complexity.

Example 1:

```
Input: secret = [-1,0,3,5,9,12], target = 9
Output: 4
Explanation: 9 exists in secret and its index is 4.
```

Example 2:

```
Input: secret = [-1,0,3,5,9,12], target = 2
Output: -1
Explanation: 2 does not exist in secret so return -1.
```

Constraints:

- $1 <= secret.length <= 10^4$
- $-10^4 <= secret[i], target <= 10^4$
- `secret` is sorted in a strictly increasing order.