https://course.acciojob.com/idle?question=1f8e91a5-e54f-488c-8706-aaf17b2ceb3d

- EASY
- Max Score: 30 Points

Binary Search

Given an sorted integer array of size `n` which contains unique elements, find the `index` (0-based) of the given `key`.

If the `key` is not present then return `-1`.

Input Format

First line contains two spaced integers the array size `n` and `key`

Next line contains `n` spaced integers.

Output Format

Return the index of the key in the array in the given function or return -1 if it's not present in the array.

Example 1

Input

```
7 730
43 210 723 730 832 838 997
```

Output

```
3
```

Explanation

According to 0 based indexing the key 730 is at index 3.

Example 2

Input

```
10 1
24 50 62 74 87 434 477 625 783 940
```

Output

```
-1
```

Explanation

key 1 is not present in the array so we return -1.

Constraints

1 <= n <= 10^5

1 <= key <= 10^9

1 <= arr[i] <= 10^9

Topic Tags

- **Binary Search**

- Arrays

My code

```
// in java
import java.util.*;

public class Main {
  public static int findIndex(int key, int[] arr) {
    //Write code here
int n=arr.length;
        int i=0,j=n-1;
        while(i<=j)
            {
            int mid=(i+j)/2;
                if(arr[mid]==key)
                    return mid;
                if(arr[mid]>key)
                    j=mid-1;
                else i=mid+1;
            }
        return -1;
  }

  public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    int n, key;
    n = sc.nextInt();
    key = sc.nextInt();
    int[] arr = new int[n];
    for (int i = 0; i < n; i++)
```

```
            arr[i] = sc.nextInt();
        System.out.println(findIndex(key, arr));
        sc.close();
    }
}
```