CHALLENGE INFORMATION

✅ You have already solved this challenge ! Though you can run the code with different logic ! ✕

Course	JAVA	**Session**	Arrays	**Question Information**		Level 1	Challenge 40
Problem							

Question description

Simon enjoys listening to music. He lives in Jonas's Town. A few days ago he had a birthday, so his parents gave him a gift: MP3-player! Simon was the happiest man in the world! Now he can listen his favorite songs whenever he wants!

Simon built up his own playlist. The playlist consists of N songs, each has a **unique** positive integer length. Simon likes all the songs from his playlist, but there is a song, which he likes more than the others. It's named "Brother Jeremiah".

After creation of the playlist, Simon decided to sort the songs in increasing order of their lengths. For example, if the lengths of the songs in playlist was {1, 3, 5, 2, 4} after sorting it becomes {1, 2, 3, 4, 5}. Before the sorting, "Brother Jeremiah" was on K-th position (1-indexing is assumed for the playlist) in the playlist.

Simon needs your help! He gives you all the information of his playlist. Your task is to find the position of "Brother Jeremiah" in the sorted playlist.

Constraints:

$1 \leq K \leq N \leq 100$

$1 \leq A_i \leq 10^9$

Input Format:

The first line of each test case contains one integer N denoting the number of songs in Simon's playlist. The second line contains N space-separated integers A1,A2,A3....An denoting the lengths of Simon's songs. The third line contains the only integer K - the position of "Brother Jeremiah" in the initial playlist.

Output Format:

For each test case, output a single line containing the position of "Brother Jeremiah" in the sorted playlist.
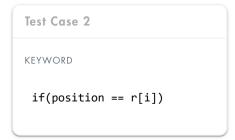
Test Cases

∨ Logical Test Cases

Test Case 1

INPUT (STDIN)

```
4
1 3 4 2
2
```

EXPECTED OUTPUT

```
3
```

Test Case 2

INPUT (STDIN)

```
5
1 2 3 9 4
1
```

EXPECTED OUTPUT

```
1
```

∨ Mandatory Test Cases

Test Case 1

KEYWORD

```
for(int i=0;i<s;i++)
```

Test Case 2

KEYWORD

```
if(position == r[i])
```

∨ Complexity Test Cases

Test Case 1	Test Case 2	Test Case 3
CYCLOMATIC COMPLEXITY	TOKEN COUNT	NLOC
5	190	27

Code Editor

Code Editor   JAVA SE 1.8   Light Theme

```java
1   import java.util.*;
2   public class Class332241010280 {
3       public static void main(String[] args) {
4       int s;
5       Scanner sc=new Scanner(System.in);
6       s=sc.nextInt();
7       int[] r=new int[s];
8
9       for(int i=0;i<s;i++){
10          r[i]=sc.nextInt();
11      }
12      int k=sc.nextInt();
13      int position=r[k-1];
14      Arrays.sort(r);
15      int p=0;
16      for(int i=0;i<s;i++){
17          if(position == r[i]){
18              p=i+1;
19              break;
20          }
21      }
22      System.out.println(p);
23      }
24  }
```

Custom Input (stdin)   T1   T2

Type Here

Output   MATCH T1   MATCH T2

Empty

Complexity Analysis

Test Case Status