 ☰ 🏠 **Practice**   🕐 **Compete**   💼 **Jobs**   🏅 **Rank**   🏆 **Leaderboard**   🔍 [_____]   💬 📢 ⬢ blakek

Badge Progress (Details)

Dashboard > Algorithms > Implementation > Circular Array Rotation   [_____]   **Points: 136.00 Rank: 231178**

Circular Array Rotation 🔖

🧑 **by darkshadows**

Problem	Submissions	Leaderboard	Discussions	Editorial

John Watson performs an operation called a _right circular rotation_ on an array of integers, $[a_0, a_1, \ldots a_{n-1}]$. After performing one _right circular rotation_ operation, the array is transformed from $[a_0, a_1, \ldots a_{n-1}]$ to $[a_{n-1}, a_0, \ldots, a_{n-2}]$.

Watson performs this operation k times. To test Sherlock's ability to identify the current element at a particular position in the rotated array, Watson asks q queries, where each query consists of a single integer, m, for which you must print the element at index m in the rotated array (i.e., the value of a_m).

Input Format

The first line contains 3 space-separated integers, n, k, and q, respectively.
The second line contains n space-separated integers, where each integer i describes array element a_i (where $0 \le i < n$).
Each of the q subsequent lines contains a single integer denoting m.

Constraints

- $1 \le n \le 10^5$
- $1 \le a_i \le 10^5$
- $1 \le k \le 10^5$
- $1 \le q \le 500$
- $0 \le m \le n - 1$

Output Format

For each query, print the value of the element at index m of the rotated array on a new line.

Sample Input 0

```
3 2 3
1 2 3
0
1
2
```

Sample Output 0

```
2
3
1
```

Explanation 0

After the first rotation, the array becomes $[3, 1, 2]$.
After the second (and final) rotation, the array becomes $[2, 3, 1]$.

Let's refer to the array's final state as array b. For each query, we just have to print the value of b_m on a new line:

1. $m = 0$, so we print 2 on a new line.

2. $m = 1$, so we print 3 on a new line.

3. $m = 2$, so we print 1 on a new line.

f ⅴ in

Submissions: 86472
Max Score: 20
Difficulty: Easy

Rate This Challenge:
☆ ☆ ☆ ☆ ☆

More

Need Help? Get advice from the discussion forum for this challenge. Or check out the environments page

---

Current Buffer (saved locally, editable) ⑂ ⏱   Go ⌄   ⤢ ⚙

```go
1  package main
2
3  import "fmt"
4
5 ▾/*n, how many numbers and the max size of those numbers less then (zero based index)
6
7  k - how many times rotates,
8
9  q - queries, each one has single integer of m
10
11 print element at index m in the rotated array
12 first line contains 3 ints,
13 n, k, q
14
15 n=3
16 k=2
17 q=3
18 second line contains n ints, where each int describes an array element
19
20 q number of more lines
21 contains single int denoting m*/
22 ▾func main() {
23
24     nvalue, kvalue, qvalue := 0, 0, 0
25
26     fmt.Scanf("%d", &nvalue)
27     fmt.Scanf("%d", &kvalue)
28     fmt.Scanf("%d", &qvalue)
29
30     arr := make([]int, nvalue, nvalue)
31     arr2 := make([]int, nvalue, nvalue)
32     qvals := make([]int, qvalue, qvalue)
33
34 ▾   for i := 0; i < nvalue; i++ {
35 ▾       fmt.Scanf("%d", &arr[i])
36     }
37     rot := kvalue % nvalue
38     newI := 0
39 ▾   for i := 0; i < nvalue; i++ {
40
41 ▾       if i >= nvalue-rot {
42 ▾           arr2[newI] = arr[i]
43           newI = newI + 1
44 ▾           /*
45           oldi := i
46           for a := 0; a < rot; a, oldi = a+1, oldi+1 {
47           arr2[a] = arr[oldi]
48
49
```

```
50              }*/
51
52 ▾       } else {
53 ▾           arr2[i+rot] = arr[i]
54          }
55
56
57      }
58
59 ▾     for i := 0; i < qvalue; i++ {
60 ▾         fmt.Scanf("%d", &qvals[i])
61
62      }
63
64 ▾     for i := 0; i < qvalue; i++ {
65 ▾         fmt.Println(arr2[qvals[i]])
66      }
67
68  }
69
```

Line: 1 Col: 13

⬆ Upload Code as File   ☐ Test against custom input   Run Code   Submit Code

Congrats, you solved this challenge!

✔ Test Case #0	✔ Test Case #1	✔ Test Case #2
✔ Test Case #3	✔ Test Case #4	✔ Test Case #5
✔ Test Case #6	✔ Test Case #7	✔ Test Case #8
✔ Test Case #9	✔ Test Case #10	✔ Test Case #11
✔ Test Case #12	✔ Test Case #13	✔ Test Case #14
✔ Test Case #15		

Next Challenge

Join us on IRC at #hackerrank on freenode for hugs or bugs.

Contest Calendar | Interview Prep | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature