

# CS1020E: DATA STRUCTURES AND ALGORITHMS I

## Lab 1 Ex 2 – Round the Ring

(Week 3, starting 22 August 2016)

Before attempting the problem, read the exercise completely. If you choose any of the 2 improvements, you only need to develop one program and submit it under 2 or 3 different tasks.

### Problem 40%

$5 \leq N \leq 650,000$  people sit in a circle blindfolded, facing each other. The first person is asked for an integer  $x_1$ , resulting in the person  $x_1$  **places clockwise** of the first person being **removed** from the circle. The person **immediately clockwise** of the eliminated person **is then asked** for the next integer  $x_2$ , deciding the second person to be removed, and so on. All  $x_i > 0$ , and will fit within a 32-bit signed int.

For example, we start off with 8 people [0, 1, 2, 3, 4, 5, 6, 7] and ask for 5 numbers. The replies given are (10, 6, 3, 5, 10). The people who are left in the circle are [0, 3, 5] because:

[0, 1, 2, 3, 4, 5, 6, 7]	0 calls 10 removing 2
[0, 1, 3, 4, 5, 6, 7]	3 calls 6 removing 1
[0, 3, 4, 5, 6, 7]	3 calls 3 removing 6
[0, 3, 4, 5, 7]	7 calls 5 removing 7 lol ouch
[0, 3, 4, 5]	0 calls 10 removing 4

As you can see in the example, the first person in the circle, before any removals are made, is labeled 0, and each person immediately clockwise then is labeled 1 higher. You are to **determine who are still in** the circle at the end of a number of removals, in **ascending order** of their labels. There will be **at least 3 people remaining** in the circle at the end.

The first line in the input contains only  $N$ . The second line contains  $x_1 x_2 x_3 \dots$

#### Sample Input

```
8
10 6 3 5 10
```

#### Sample Output

```
[0 3 5]
```

#### Submission

Your source file should be named round\_the\_ring.cpp

Submit to task Ex2: Round The Ring (**Ordinary**)

### First Improvement



+15%

Solve the problem for large  $N$  **within 3 seconds**, given that:

"When the number of people in the circle is still very large, after counting  $x_i$  places clockwise, the person to be removed will usually end up a few places clockwise of the person we start counting from."

If you can remove each person efficiently, are able to skip people already removed, and do not unnecessarily move round the circle, you should be able to get an accepted solution for this part.

Submit to **BOTH** tasks:

- Ex2: Round The Ring (Ordinary)
- Ex2: Round The Ring (**First Improvement**)

### Second Improvement



+5%

Solve the problem for large  $N$  within 3 seconds, **even for large  $x_i$** . To accomplish that, counting  $x$  places clockwise and removing one person should require visiting at most proportional to  $\log_2 N$  people.

If you manage to do get an accepted solution for this part, submit that solution for all three tasks. Your solution will be accepted for the "first improvement" task, and you will get both bonuses.

Submit to **ALL THREE** tasks:

- Ex2: Round The Ring (Ordinary)
- Ex2: Round The Ring (First Improvement)
- Ex2: Round The Ring (**Second Improvement**)

### Can't Decide?

The first improvement is doable, with careful thought, tracing, algorithm design and testing. If you can get an accepted solution to the second improvement, you should probably not be taking this module... =p

If you are unsure whether you have an idea for the first improvement, just attempt the basic problem first. You are reminded that we will only take **your score from the latest submission** in each part.

- End of Lab 1 Ex 2 -