

# Pseudorandom Repetition

I've been working on a program that needs lots of random numbers, so I wrote a Linear Congruential Generator to make them. A LCG is a kind of pseudorandom number generator where the next value  $n_{i+1}$  equals  $a$  times the previous  $n$ ,  $n_i$ , plus  $c$ , mod  $m$ .

$$n_{i+1} = an_i + c \pmod{m}$$

I picked a LCG because it was simple, but LCGs eventually repeat themselves. Given the values  $a$ ,  $c$ ,  $m$ , and the initial seed  $n_i$ , where all four numbers are  $> 0$  and  $< 2^{63} - 1$ , compute the period of the generator, as well as the first value to repeat.

## Input Format

The input will consist of four integers,  $a$ ,  $c$ ,  $m$ , and  $n_i$ , representing the scalar, constant, modulus, and initial seed, respectively.

## Output Format

Your output be the period,  $p$  of the given parameters as well as  $n_{i+x}$ , the first number to be repeated. Expect the inputs as well as the outputs to be quite large.

## Sample Input

```
97
1
109
5
```

## Sample Output

```
28
5
```