Linear Congruential Generator Problem

Input file:	standard input
Output file:	standard output
Time limit:	2 seconds
Memory limit:	1024 megabytes

Consider the following C++ code to generate a pseudorandom permutation.

```
long long x, a, b, p;

int n;
long long rand() {
        x = ((__int128)x * a + b) % p;
        return x;
}
int main() {
    cin >> n;
    cin >> x >> a >> b >> p;
    for (int i = 1; i <= n; i++) { //random shuffle [1, 2,..., n]
        perm[i] = i;
        swap(perm[i], perm[rand() % i + 1]);
    }
    for (int i = 1; i <= n; i++) { //print the result
        cout << perm[i] << (i == n ? '\n' : ' ');
    }
}
```

You are given n, a, b, p, and the output $perm_1, perm_2, \ldots, perm_n$, find x $(0 \le x < p)$.

Input

In the first line, n $(n = 10^5)$.

In the second line, $perm_1, perm_2, \ldots, perm_n$ $(1 \le perm_i \le n)$.

In the third line, a, b, p $(2 \le a < p, 0 \le b < p, 500 \le p \le 10^{16})$. p is a prime, a, b, x are chosen uniformly at random from their respective ranges.

Output

One integer — x. If there are multiple solutions, you may output any.

Example

standard input	standard output
100000 38898 35776 38192 80605 84959 ... 356846063 184710711 911417497 (truncated)	681779867

Note

The full sample test case is available as an attachment in the contest system.