Problem A. Sliding Window Sum

Time Limit	1000 ms
Mem Limit	524288 kB

You are given an array of n integers. Your task is to calculate the sum of each window of k elements, from left to right.

In this problem the input data is large and it is created using a generator.

Input

The first line contains two integers n and k: the number of elements and the size of the window.

The next line contains four integers x, a, b and c: the input generator parameters. The input is generated as follows:

- $x_1 = x$
- $x_i = (ax_{i-1} + b) \bmod c$ for $i = 2, 3, \ldots, n$

Output

Print the xor of all window sums.

Constraints

- $1 \leq k \leq n \leq 10^7$
- $0 \leq x, a, b \leq 10^9$
- $1 \leq c \leq 10^9$

Example

Input	Output
8 5 3 7 1 11	12

Explanation: The input array is $[3, 0, 1, 8, 2, 4, 7, 6]$. The windows are $[3, 0, 1, 8, 2]$, $[0, 1, 8, 2, 4]$, $[1, 8, 2, 4, 7]$ and $[8, 2, 4, 7, 6]$, and their sums are 14, 15, 22 and 27. Thus, the answer is $14 \oplus 15 \oplus 22 \oplus 27 = 12$.

Problem B. Sliding Window Minimum

Time Limit 1000 ms

Mem Limit 524288 kB

You are given an array of n integers. Your task is to calculate the minimum of each window of k elements, from left to right.

In this problem the input data is large and it is created using a generator.

Input

The first line contains two integers n and k: the number of elements and the size of the window.

The next line contains four integers x, a, b and c: the input generator parameters. The input is generated as follows:

- $x_1 = x$
- $x_i = (ax_{i-1} + b) \bmod c$ for $i = 2, 3, \ldots, n$

Output

Print the xor of all window minimums.

Constraints

- $1 \leq k \leq n \leq 10^7$
- $0 \leq x, a, b \leq 10^9$
- $1 \leq c \leq 10^9$

Example

Input	Output
8 5 3 7 1 11	3

Explanation: The input array is $[3, 0, 1, 8, 2, 4, 7, 6]$. The windows are $[3, 0, 1, 8, 2]$, $[0, 1, 8, 2, 4]$, $[1, 8, 2, 4, 7]$ and $[8, 2, 4, 7, 6]$, and their minimums are 0, 0, 1 and 2. Thus, the answer is $0 \oplus 0 \oplus 1 \oplus 2 = 3$.

Problem C. Sliding Window Xor

Time Limit	1000 ms
Mem Limit	524288 kB

You are given an array of n integers. Your task is to calculate the bitwise xor of each window of k elements, from left to right.

In this problem the input data is large and it is created using a generator.

Input

The first line contains two integers n and k: the number of elements and the size of the window.

The next line contains four integers x, a, b and c: the input generator parameters. The input is generated as follows:

- $x_1 = x$
- $x_i = (ax_{i-1} + b) \bmod c$ for $i = 2, 3, \ldots, n$

Output

Print the xor of all window xors.

Constraints

- $1 \leq k \leq n \leq 10^7$
- $0 \leq x, a, b \leq 10^9$
- $1 \leq c \leq 10^9$

Example

Input	Output

`8 5`
`3 7 1 11`
	`0`

Explanation: The input array is $[3, 0, 1, 8, 2, 4, 7, 6]$. The windows are $[3, 0, 1, 8, 2]$, $[0, 1, 8, 2, 4]$, $[1, 8, 2, 4, 7]$ and $[8, 2, 4, 7, 6]$, and their xors are 8, 15, 8 and 15. Thus, the answer is $8 \oplus 15 \oplus 8 \oplus 15 = 0$.

Problem D. Sliding Window Or

Time Limit	1000 ms
Mem Limit	524288 kB

You are given an array of n integers. Your task is to calculate the bitwise or of each window of k elements, from left to right.

In this problem the input data is large and it is created using a generator.

Input

The first line contains two integers n and k: the number of elements and the size of the window.

The next line contains four integers x, a, b and c: the input generator parameters. The input is generated as follows:

- $x_1 = x$
- $x_i = (ax_{i-1} + b) \bmod c$ for $i = 2, 3, \ldots, n$

Output

Print the xor of all window ors.

Constraints

- $1 \leq k \leq n \leq 10^7$
- $0 \leq x, a, b \leq 10^9$
- $1 \leq c \leq 10^9$

Example

Input	Output
8 5 3 7 1 11	4

Explanation: The input array is $[3, 0, 1, 8, 2, 4, 7, 6]$. The windows are $[3, 0, 1, 8, 2]$, $[0, 1, 8, 2, 4]$, $[1, 8, 2, 4, 7]$ and $[8, 2, 4, 7, 6]$, and their ors are 11, 15, 15 and 15. Thus, the answer is $11 \oplus 15 \oplus 15 \oplus 15 = 4$.

Problem E. Sliding Window Distinct Values

Time Limit 1000 ms
Mem Limit 524288 kB

You are given an array of n integers. Your task is to calculate the number of distinct values in each window of k elements, from left to right.

Input

The first line contains two integers n and k: the number of elements and the size of the window.

Then there are n integers $x_1, x_2, \ldots, x_n$: the contents of the array.

Output

Print $n - k + 1$ values: the numbers of distinct values.

Constraints

- $1 \leq k \leq n \leq 2 \cdot 10^5$
- $1 \leq x_i \leq 10^9$

Example

Input	Output
8 3 1 2 3 2 5 2 2 2	3 2 3 2 2 1

Problem F. Sliding Window Mode

Time Limit	1000 ms
Mem Limit	524288 kB

You are given an array of n integers. Your task is to calculate the mode each window of k elements, from left to right.

The mode is the most frequent element in an array. If there are several possible modes, choose the smallest of them.

Input

The first line contains two integers n and k: the number of elements and the size of the window.

Then there are n integers $x_1, x_2, \dots, x_n$: the contents of the array.

Output

Print $n - k + 1$ values: the modes.

Constraints

- $1 \le k \le n \le 2 \cdot 10^5$
- $1 \le x_i \le 10^9$

Example

Input	Output

`8 3`
`1 2 3 2 5 2 4 4`
	`1 2 2 2 2 4`

Problem G. Sliding Window Mex

Time Limit	1000 ms
Mem Limit	524288 kB

You are given an array of n integers. Your task is to calculate the mex of each window of k elements, from left to right.

The mex is the smallest nonnegative integer that does not appear in the array. For example, the mex for $[3, 1, 4, 3, 0, 5]$ is 2.

Input

The first line contains two integers n and k: the number of elements and the size of the window.

Then there are n integers $x_1, x_2, \ldots, x_n$: the contents of the array.

Output

Print $n - k + 1$ values: the mex values.

Constraints

- $1 \le k \le n \le 2 \cdot 10^5$
- $0 \le x_i \le 10^9$

Example

Input	Output

`8 3`
`1 2 1 0 5 1 1 0`
	`0 3 2 2 0 2`

Problem H. Sliding Window Median

Time Limit 1000 ms
Mem Limit 524288 kB

You are given an array of n integers. Your task is to calculate the median of each window of k elements, from left to right.

The median is the middle element when the elements are sorted. If the number of elements is even, there are two possible medians and we assume that the median is the smaller of them.

Input

The first line contains two integers n and k: the number of elements and the size of the window.

Then there are n integers $x_1, x_2, \ldots, x_n$: the contents of the array.

Output

Print $n - k + 1$ values: the medians.

Constraints

- $1 \leq k \leq n \leq 2 \cdot 10^5$
- $1 \leq x_i \leq 10^9$

Example

Input	Output
8 3 2 4 3 5 8 1 2 1	3 4 5 5 2 1

Problem I. Sliding Window Cost

Time Limit 1000 ms

Mem Limit 524288 kB

You are given an array of n integers. Your task is to calculate for each window of k elements, from left to right, the minimum total cost of making all elements equal.

You can increase or decrease each element with cost x where x is the difference between the new and the original value. The total cost is the sum of such costs.

Input

The first line contains two integers n and k: the number of elements and the size of the window.

Then there are n integers $x_1, x_2, \dots, x_n$: the contents of the array.

Output

Output $n - k + 1$ values: the costs.

Constraints

- $1 \le k \le n \le 2 \cdot 10^5$
- $1 \le x_i \le 10^9$

Example

Input	Output
8 3 2 4 3 5 8 1 2 1	2 2 5 7 7 1

Problem J. Sliding Window Inversions

Time Limit 1000 ms

Mem Limit 524288 kB

You are given an array of n integers. Your task is to calculate the number of inversions in each window of k elements, from left to right.

An inversion is a pair of elements where the left element is larger than the right element.

Input

The first line contains two integers n and k: the number of elements and the size of the window.

Then there are n integers $x_1, x_2, \ldots, x_n$: the contents of the array.

Output

Print $n - k + 1$ values: the numbers of inversions.

Constraints

- $1 \leq k \leq n \leq 2 \cdot 10^5$
- $1 \leq x_i \leq 10^9$

Example

Input	Output
8 3 1 2 3 2 5 2 4 4	0 1 1 1 2 0

Problem K. Sliding Window Advertisement

Time Limit 1000 ms
Mem Limit 524288 kB

A fence consists of n vertical boards. The width of each board is 1 and their heights may vary.

You want to attach a rectangular advertisement to the fence. Your task is to calculate the maximum area of such an advertisement in each window of k vertical boards, from left to right.

Input

The first line contains two integers n and k: the width of the fence and the size of the window.

After this, there are n integers $x_1, x_2, \ldots, x_n$: the height of each board.

Output

Print $n - k + 1$ integers: the maximum areas of the advertisements.

Constraints

- $1 \leq k \leq n \leq 2 \cdot 10^5$
- $1 \leq x_i \leq 10^9$

Example

Input	Output
8 3 4 1 5 3 3 2 4 1	5 6 9 6 6 4