Java Dequeue

In computer science, a double-ended queue (dequeue, often abbreviated to deque, pronounced deck) is an abstract data type that generalizes a queue, for which elements can be added to or removed from either the front (head) or back (tail).

Deque interfaces can be implemented using various types of collections such as `LinkedList` or `ArrayDeque` classes. For example, deque can be declared as:

```
Deque deque = new LinkedList<>();
or
Deque deque = new ArrayDeque<>();
```

You can find more details about Deque here.

In this problem, you are given N integers. You need to find the maximum number of unique integers among all the possible contiguous subarrays of size M.

Note: Time limit is 3 second for this problem.

Input Format

The first line of input contains two integers N and M: representing the total number of integers and the size of the subarray, respectively. The next line contains N space separated integers.

Constraints

$1 \leq N \leq 100000$
$1 \leq M \leq 100000$
$M \leq N$
The numbers in the array will range between $[0, 10000000]$.

Output Format

Print the *maximum* number of unique integers among all possible contiguous subarrays of size M.

Sample Input

```
6 3
5 3 5 2 3 2
```

Sample Output

```
3
```

Explanation

In the sample testcase, there are 4 subarrays of contiguous numbers.

$s1 = \langle 5, 3, 5 \rangle$ - Has 2 unique numbers.

$s2 = \langle 3, 5, 2 \rangle$ - Has 3 unique numbers.

$s3 = \langle 5, 2, 3 \rangle$ - Has 3 unique numbers.

$s4 = \langle 2, 3, 2 \rangle$ - Has 2 unique numbers.

In these subarrays, there are $2, 3, 3, 2$ unique numbers, respectively. The maximum amount of unique numbers among all possible contiguous subarrays is 3.