**Java Dequeue**

<https://www.hackerrank.com/challenges/java-dequeue/problem>

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](http://docs.oracle.com/javase/7/docs/api/java/util/Deque.html).

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 <= N <= 100000*

*1 <= M <= 100000*

*M <= 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 = (5, 3, 5)* - Has *2* unique numbers.

*s2 = (3, 5, 2)* - Has *3* unique numbers.

*s3 = (5, 2, 3)* - Has *3* unique numbers.

*s4 = (2, 3, 2)* - 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*.