

Project #4: Longest sequence of close numbers

Task description:

Given a sequence of N numbers in non-decreasing order, find the size of the longest subsequence composed of consecutive elements of the input sequence whose difference between the maximum and minimum elements in it is below a threshold T .

For instance, if the input contains the following sequence of 10 numbers:

1 2 3 3 3 4 4 5 6 6

All the following are valid subsequences of consecutive elements from the input sequence above:

1 2

1 2 3 3

3 3 4 4 5 6

3 3 3 4 4

The differences between the maximum and minimum elements in the subsequences above are respectively **1** ($2-1=1$), **2** ($3-1=2$), **3** ($6-3=3$) and **1** ($4-3=1$).

For $T = 2$, the longest subsequence of consecutive elements in the input above starts at the third element and ends in the seventh element of the original sequence (the last subsequence in the example above). It has size 5 and contains the elements "3 3 3 4 4". The difference between the maximum and minimum elements in this subsequence is 1, which is below T . Other subsequences also have the difference between the maximum and minimum elements below T (see the first subsequence in the example above), but their size are smaller than 5.

You are not allowed to use array subscripting in this project, not even while reading the input data. **In other words, you can only use brackets (`[` and `]`) to define the size of the arrays (variable declaration). If you use them in any other location, there will be grade deductions.**

Input specification:

The input of a test case contains two lines. The first line has two integers, N and T , indicating the size of the input sequence and the threshold value, respectively ($1 \leq N \leq 1000$, $1 \leq T \leq 10^6$). The second line contains N numbers in non-decreasing order separated by space. All numbers in the sequence are non-negative integers smaller than or equal to 10^6 .

Output specification:

Your program must print a single line as output with the size of the longest subsequence found. Do not forget the new-line character in the end.

Example #1:

Input	Output
10 2 1 2 3 3 3 4 4 5 6 6	5

Example #2:

Input	Output
10 1 1 2 3 3 3 4 4 5 6 6	3

Example #3:

Input	Output
15 1000 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	15