Problem G
Bounded Jump

You are given a sequence of integers $S_{1..N}$, your task is to find the length of the longest L-K-bounded jump of S.

A sequence of integers of length m $(a_1, a_2, ..., a_m)$ is called an L-K-bounded jump of S if it satisfies the following properties.

(i) $1 \leq a_i \leq N$ for all $1 \leq i \leq m$

(ii) $a_i < a_{i+1} \leq a_i + L$ for all $1 \leq i < m$

(iii) The difference between S_{a_i} and $S_{a_{i+1}}$ is no more than K for all $1 \leq i < m$

For example, let $S_{1..7} = \{3, 1, 4, 5, 3, 2, 5\}$ while $L = 2$ and $K = 1$. There are several 2-1-bounded jump can be found in S, e.g., $(1, 3, 4)$, $(1, 3, 5, 6)$, $(2, 6)$, $(3, 4)$, etc. The longest 2-1-bounded jump in this example is $(1, 3, 5, 6)$ with a length of 4. Also, note that $(3, 7)$ and $(1, 2)$ are not a 2-1-bounded jump. The former violates property (ii) while the latter violates property (iii).

Input

Input begins with an integer T $(1 \leq T \leq 1000)$ representing the number of cases.

Each case begins with three integers N L K $(1 \leq L \leq N \leq 50\,000;\ 0 \leq K \leq 100\,000)$. The next line contains N integers S_i $(0 \leq S_i \leq 10^9)$ representing the given sequence.

It is guaranteed that the sum of N over all cases does not exceed $500\,000$.

Output

For each case, output in a line "`Case #X: Y`" (without quotes) where `X` is the case number (starts from 1) and `Y` is the output for the respective case.

Sample Input #1

```
4
7 2 1
3 1 4 5 3 2 5
5 1 5
50 40 55 53 49
6 6 0
17 13 20 8 13 25
7 4 10
1 2 3 4 3 2 1
```

Sample Output #1

```
Case #1: 4
Case #2: 3
Case #3: 2
Case #4: 7
```

Explanation for the sample input/output #1

For the 2^{nd} case, the longest 1-5-bounded jump is $(3, 4, 5)$.

For the 3^{rd} case, the longest 6-0 bounded jump is $(2, 5)$.

For the 4^{th} case, the longest 4-10 bounded jump is $(1, 2, 3, 4, 5, 6, 7)$.