I. Photo Processing

Evlampiy has found one more cool application to process photos. However the application has certain limitations.

Each photo i has a contrast v_i. In order for the processing to be truly of high quality, the application must receive at least k photos with contrasts which differ as little as possible.

Evlampiy already knows the contrast v_i for each of his n photos. Now he wants to split the photos into groups, so that each group contains at least k photos. As a result, each photo must belong to exactly one group.

He considers a processing time of the j-th group to be the difference between the maximum and minimum values of v_i in the group. Because of multithreading the processing time of a division into groups is the maximum processing time among all groups.

Split n photos into groups in a such way that the processing time of the division is the minimum possible, i.e. that the the maximum processing time over all groups as least as possible.

Input
The first line contains two integers n and k ($1 \leq k \leq n \leq 3 \cdot 10^5$) — number of photos and minimum size of a group.

The second line contains n integers $v_1, v_2, ..., v_n$ ($1 \leq v_i \leq 10^9$), where v_i is the contrast of the i-th photo.

Output
Print the minimal processing time of the division into groups.

Examples

input
5 2 50 110 130 40 120

output
20

input
4 1 2 3 4 1

output
0

Note
In the first example the photos should be split into 2 groups: $[40, 50]$ and $[110, 120, 130]$. The processing time of the first group is 10, and the processing time of the second group is 20. Maximum among 10 and 20 is 20. It is impossible to split the photos into groups in a such way that the processing time of division is less than 20.

In the second example the photos should be split into four groups, each containing one photo. So the minimal possible processing time of a division is 0.