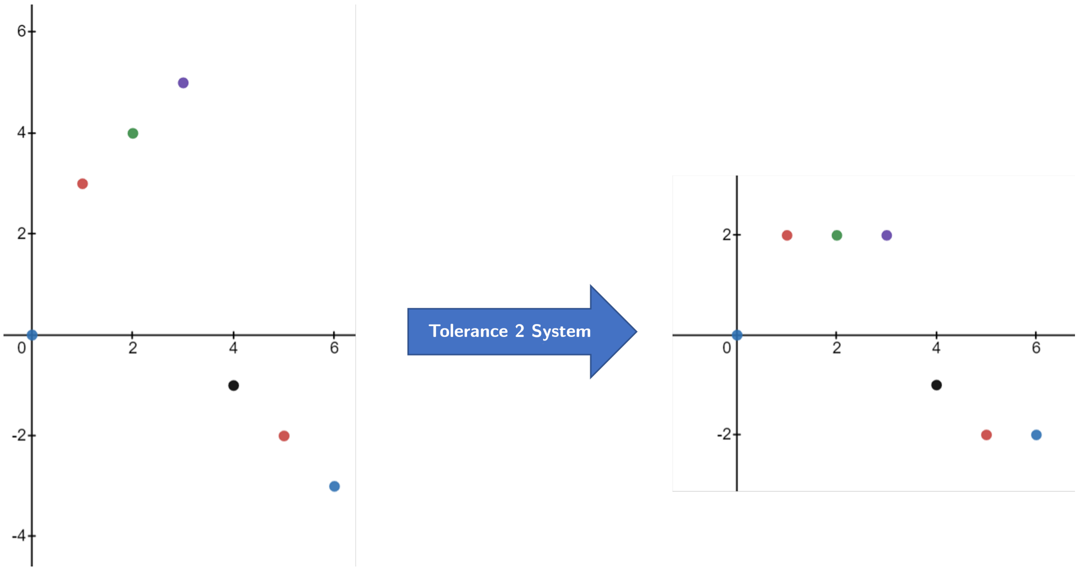
Plane Turbulences (`turbulences`)

Brianair, a reputed airline, is conducting a study about turbulences. They are working to find the optimal stabilising system for their aircraft. This is to be in place during all the flight except landing and take-off, that is, the part of the flight where the plane is supposed to fly *in a straight line*.

A stabilising system of *tolerance* x will ensure that the plane does not deviate from its desired altitude (the one it would have if it were flying in a straight line at constant altitude) by an absolute difference higher than x. It is possible to know in advance the altitude of the plane at each minute of the trip if we do not equip it with a stabilising system. You will be given all these height deviation predictions $A_0, \ldots, A_{N-1}$ for the duration of the trip N, in chronological order.

The following example shows how a stabilising system of tolerance 2 takes a flight with deviation predictions $A_0 = 0, A_1 = 3, A_2 = 4, A_3 = 5, A_4 = -1, A_5 = -2, A_6 = -3$ to a flight with actual deviations $B_0 = 0, B_1 = 2, B_2 = 2, B_3 = 2, B_4 = -1, B_5 = -2, B_6 = -2$.



Altitudes before and after applying a stabilising system with tolerance 2.

Brianair knows that customers love high-flying travels, so the customer satisfaction (i.e. the airline's gain from implementing the system) after flying on a plane with a stabilising system of tolerance x equals $\sum_{i=0}^{N-1} B_i$, where B_i is the stabilised altitude at time i. That is, $B_i = \text{sign}(A_i) \cdot \min(|A_i|, x)$.

But the cost of bribing regulators to allow a system with tolerance x equals Kx, where K is some nonnegative constant. The airline therefore wants to maximise its revenue from the flight, i.e. $\left(\sum_{i=0}^{N-1} B_i\right) - Kx$.

Given K and $A_0, \ldots, A_{N-1}$, would you be able to find the maximum revenue that can be attained by setting the optimal tolerance $x \geq 0$?

Implementation

You will have to submit a single `.cpp` source file.

> ☞ Among this task's attachments you will find a template `turbulences.cpp` with a sample implementation.

You have to implement the following function:

C++	`long long revenue(int N, int K, vector<long long> A);`

- Integer N represents the duration of the flight.
- Integer K represents the cost coefficient.
- The array A, indexed from 0 to $N-1$, contains the values $A_0, A_1, \ldots, A_{N-1}$, where A_i is the predicted altitude at time i.
- The function should return the maximum revenue that can be obtained.

The grader will call the function `revenue` and will print its return value to the output file.

Sample Grader

The task's directory contains a simplified version of the jury grader, which you can use to test your solution locally. The simplified grader reads the input data from `stdin`, calls the functions that you must implement, and finally writes the output to `stdout`.

The input is made up of 2 lines, containing:

- Line 1: the integers N and K.
- Line 2: the integers A_i, separated by spaces.

The output is made up of a single line, containing the value returned by the function `revenue`.

Constraints

- $1 \le N \le 2 \times 10^5$.
- $0 \le K \le 2 \times 10^5$.
- $-10^{12} \le A_i \le 10^{12}$.

Scoring

Your program will be tested on a set of test cases grouped by subtask. To obtain the score associated to a subtask, you need to correctly solve all the test cases it contains.

- **Subtask 1** [**0 points**]: Sample test cases.
- **Subtask 2** [**15 points**]: $N = 1$.
- **Subtask 3** [**30 points**]: $N \le 10^2$, $K \le 10^2$, $-10^2 \le A_i \le 10^2$ for each $i = 0, \ldots, N-1$.
- **Subtask 4** [**17 points**]: All A_i are equal.
- **Subtask 5** [**18 points**]: All A_i are nonnegative.
- **Subtask 6** [**20 points**]: No additional constraints.

Examples

stdin	stdout
7 1 0 3 4 5 -1 -2 -3	1
5 1 7 8 -2 5 -10	3
5 0 1000000000000 1000000000000 1000000000000 1000000000000 1000000000000	5000000000000

Explanation

In the **first sample case**, the situation is as described in the picture above. The optimal revenue is obtained with $x = 5$.

In the **second sample case**, the optimal revenue can be obtained by setting $x = 5$. Therefore, the total revenue is $(5 + 5 + -2 + 5 + -5) - 1 \cdot 5 = 3$.

In the **third sample case**, the optimal revenue can be obtained by setting any $x \geq 10^{12}$.