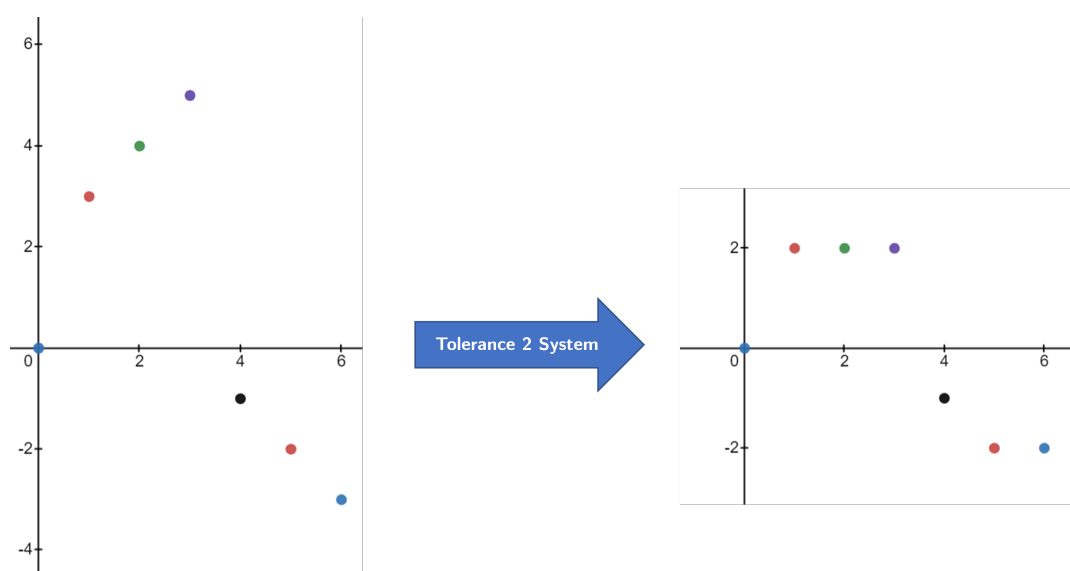


Plane Turbulences (turbulences)

Brianair, eine renommierte Fluggesellschaft, führt eine Studie über Turbulenzen durch. Sie arbeitet daran, das optimale Stabilisierungssystem für ihr Flugzeug zu finden. Dieses System soll während des gesamten Fluges außer beim Start und bei der Landung eingesetzt werden, d. h. während des Teils des Fluges, in dem das Flugzeug geradeaus fliegen soll *in einer geraden Linie*.

Ein Stabilisierungssystem mit *Toleranz* x sorgt dafür, dass das Flugzeug nicht von seiner gewünschten Höhe abweicht (die es hätte, wenn es in einer geraden Linie mit konstanter Höhe fliegen würde) um eine absolute Differenz grösser als x abweicht. Es ist möglich, die Höhe des Flugzeugs in jeder Minute der Reise im Voraus zu kennen, wenn wir es nicht mit einem Stabilisierungssystem ausstatten. Sie erhalten alle diese Höhenabweichungsvorhersagen A_0, \dots, A_{N-1} für die Dauer der Reise N , in chronologischer Reihenfolge.

Das folgende Beispiel zeigt, wie ein Stabilisierungssystem der Toleranz 2 einen Flug mit Abweichungsvorhersagen $A_0 = 0, A_1 = 3, A_2 = 4, A_3 = 5, A_4 = -1, A_5 = -2, A_6 = -3$ zu einem Flug mit tatsächlichen Abweichungen $B_0 = 0, B_1 = 2, B_2 = 2, B_3 = 2, B_4 = -1, B_5 = -2, B_6 = -2$.



Höhenangaben vor und nach Anwendung eines Stabilisierungssystems mit Toleranz 2.

Brianair weiss, dass die Kunden gerne hoch fliegen, daher ist die Kundenzufriedenheit (d. h. der Gewinn der Fluggesellschaft durch die Implementierung des Systems) nach einem Flug in einem Flugzeug mit einem Stabilisierungssystem der Toleranz x gleich $\sum_{i=0}^{N-1} B_i$, wobei B_i die stabilisierte Höhe zum Zeitpunkt i ist. Das heisst, $B_i = \text{sign}(A_i) \cdot \min(|A_i|, x)$.

Die Kosten für die Besteuerung von Regulierungsbehörden, um ein System mit der Toleranz x zuzulassen, sind jedoch gleich Kx , wobei K eine nichtnegative Konstante ist. Die Fluggesellschaft möchte daher ihre Einnahmen aus dem Flug maximieren, d.h. $\left(\sum_{i=0}^{N-1} B_i\right) - Kx$.

Kannst du unter der Voraussetzung von K und A_0, \dots, A_{N-1} den maximalen Ertrag finden, der durch die Einstellung der optimalen Toleranz $x \geq 0$ erreicht werden kann?

Implementierung

Du musst eine einzige .cpp-Quelltextdatei einreichen.

📖 In den Anhängen zu dieser Aufgabe findest du eine Vorlage `turbulences.cpp` mit einer Beispielimplementierung.

Du musst die folgende Funktion implementieren:

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

- Die ganze Zahl N steht für die Dauer des Fluges.
- Die ganze Zahl K steht für den Kostenkoeffizienten.
- Das Array A , indiziert von 0 bis $N - 1$, enthält die Werte A_0, A_1, \dots, A_{N-1} , wobei A_i die vorhergesagte Höhe zum Zeitpunkt i ist.
- Die Funktion soll den maximal erzielbaren Ertrag zurückgeben.

Der Grader ruft die Funktion `revenue` auf und druckt ihren Rückgabewert Wert in die Ausgabedatei.

Beispielgrader

Das Verzeichnis der Aufgabe enthält eine vereinfachte Version des Jury-Graders, die du verwenden kannst, um deine Lösung lokal zu testen. Der vereinfachte Grader liest die Eingabedaten aus `stdin`, ruft die Funktionen auf, die du implementieren musst, und schreibt schliesslich die Ausgabe in `stdout`.

Die Eingabe besteht aus 2 Zeilen, die Folgendes enthalten:

- Zeile 1: die ganzen Zahlen N und K .
- Zeile 2: die ganzen Zahlen A_i , getrennt durch Leerzeichen.

Die Ausgabe besteht aus einer einzigen Zeile, die den Wert enthält, der von der Funktion `revenue`.

Einschränkungen

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

Punktevergabe

Dein Programm wird mit einer Reihe von Testfällen getestet, die nach Teilaufgaben gruppiert sind. Um um die einer Teilaufgabe zugeordnete Punktzahl zu erhalten, musst du alle darin enthaltenen Testfälle richtig lösen.

- **Teilaufgabe 1 [0 Punkte]:** Beispiel-Testfälle.
- **Teilaufgabe 2 [15 Punkte]:** $N = 1$.
- **Teilaufgabe 3 [30 Punkte]:** $N \leq 10^2$, $K \leq 10^2$, $-10^2 \leq A_i \leq 10^2$ für jede $i = 0, \dots, N - 1$.
- **Teilaufgabe 4 [17 Punkte]:** Alle A_i sind gleich.
- **Teilaufgabe 5 [18 Punkte]:** Alle A_i sind nichtnegativ.
- **Teilaufgabe 6 [20 Punkte]:** Keine zusätzlichen Einschränkungen.

Beispiele

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

Erklärung

Im **ersten Testfall** ist die Situation wie in der Abbildung oben beschrieben. Den optimalen Ertrag erhält man mit $x = 5$.

Im **zweiten Testfall** kann der optimale Ertrag durch die Einstellung $x = 5$ erzielt werden. Der Gesamtertrag ist also $(5 + 5 + -2 + 5 + -5) - 1 \cdot 5 = 3$.

Im **dritten Testfall** kann der optimale Ertrag durch Setzen eines beliebigen $x \geq 10^{12}$ erzielt werden.