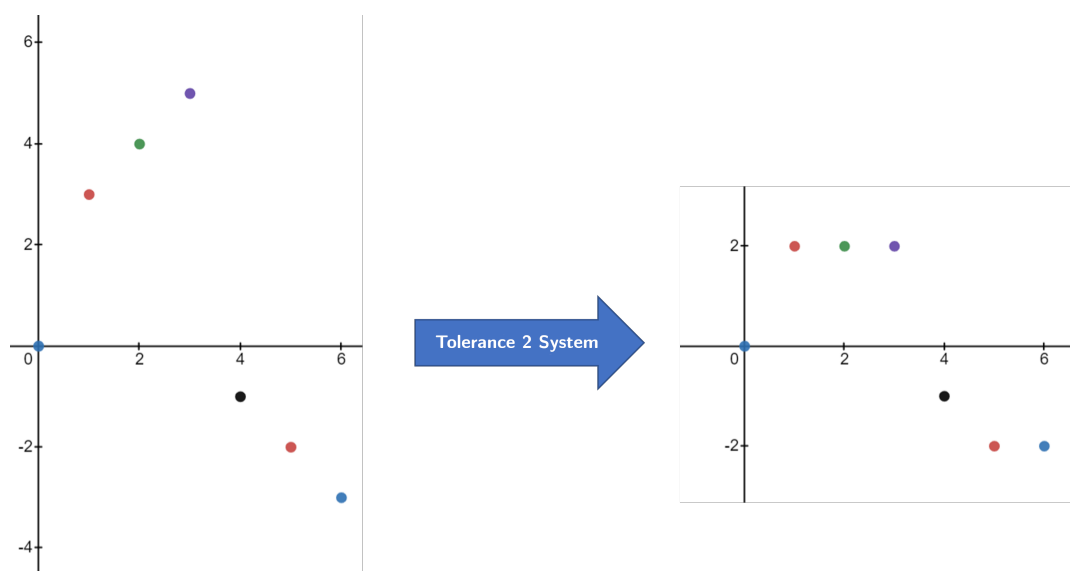


## Plane Turbulences (turbulences)

Brianair, une compagnie aérienne réputée conduit une étude sur les turbulences. Ils travaillent à trouver le système de stabilisation optimal pour leur avion. Ce système sera actif pendant toute la durée du vol sauf l'atterrissage et le décollage, c'est-à-dire la partie du vol où l'avion est censé voler *en ligne droite*.

Un système de stabilisation de *tolérance*  $x$  va s'assurer que l'avion ne dévie pas de son altitude souhaitée (celle qu'il aurait si il volait en ligne droite à altitude constante) par une différence absolue plus grande que  $x$ . Il est possible de connaître à l'avance l'altitude de l'avion à chaque minute du voyage si on ne l'équipe pas d'un système de stabilisation. On vous fournit toutes les prédictions de déviation de hauteur  $A_0, \dots, A_{N-1}$  pour la durée du voyage  $N$ , dans l'ordre chronologique.

L'exemple suivant montre comment un système de stabilisation de tolérance 2 modifie un vol avec des prédictions de déviation  $A_0 = 0, A_1 = 3, A_2 = 4, A_3 = 5, A_4 = -1, A_5 = -2, A_6 = -3$  en un vol où les déviations effectives sont  $B_0 = 0, B_1 = 2, B_2 = 2, B_3 = 2, B_4 = -1, B_5 = -2, B_6 = -2$ .



Altitudes avant et après application d'un système de stabilisation de tolérance 2.

Brianair sait que les clients aiment les voyages à haute altitude, donc la satisfaction d'un client (c'est-à-dire le gain de la compagnie aérienne à implementer le système) après avoir volé dans un avion avec un système de stabilisation de tolérance  $x$  vaut  $\sum_{i=0}^{N-1} B_i$ , où  $B_i$  est l'altitude stabilisée au temps  $i$ . Autrement dit,  $B_i = \text{sign}(A_i) \cdot \min(|A_i|, x)$ .

Toutefois, le coût pour soudoyer les régulateurs afin d'autoriser un système de tolérance  $x$  vaut  $Kx$ , où  $K$  est une constante positive. La compagnie aérienne veut donc maximiser son revenu pour le vol, soit  $\left(\sum_{i=0}^{N-1} B_i\right) - Kx$ .

Étant donnés  $K$  et  $A_0, \dots, A_{N-1}$ , veuillez trouver le revenu maximum qui puisse être obtenu en choisissant la tolérance optimale  $x \geq 0$ .

## Implémentation

Vous devez soumettre un unique fichier source `.cpp`.

📖 Parmi les fichiers joints du problème, vous trouverez un squelette de code `turbulences.cpp` avec un exemple d'implémentation.

Vous devez implémenter la fonction suivante :

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

- L'entier  $N$  représente la durée du vol.
- L'entier  $K$  représente le coefficient de coût.
- Le tableau  $A$ , indexé de 0 à  $N - 1$ , contient les valeurs  $A_0, A_1, \dots, A_{N-1}$ , où  $A_i$  est l'altitude prédite au temps  $i$ .
- La fonction doit renvoyer le revenu maximum qui peut être obtenu.

L'évaluateur va appeler la fonction `revenue` et va mettre sa valeur de retour dans le fichier de sortie.

## Évaluateur

Le dossier du problème contient une version simplifiée de l'évaluateur du jury que vous pouvez utiliser pour tester votre solution localement. Cet évaluateur simplifié lit les données en entrée du fichier `stdin`, appelle les fonction que vous devez implémenter et enfin écrit la sortie dans `stdout`.

L'entrée est constituée de  $N + 1$  lignes, contenant :

- Ligne 1 : les entiers  $N$  et  $K$ .
- Ligne 2 : les entiers  $A_i$ , séparés par des espaces.

La sortie est constituée d'une unique ligne, contenant la valeur renvoyée par la fonction `revenue`.

## Contraintes

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

## Score

Votre programme sera testé sur un ensemble de tests groupés par sous-tâche. Pour obtenir un score associé à une sous-tâche, vous devez résoudre correctement tous les tests qu'elle contient.

- **Sous-tâche 1 [ 0 points]**: Sample test cases.
- **Sous-tâche 2 [15 points]**:  $N = 1$ .
- **Sous-tâche 3 [30 points]**:  $N \leq 10^2$ ,  $K \leq 10^2$ ,  $-10^2 \leq A_i \leq 10^2$  pour chaque  $i = 0, \dots, N - 1$ .
- **Sous-tâche 4 [17 points]**: Tous les  $A_i$  sont égaux.
- **Sous-tâche 5 [18 points]**: Tous les  $A_i$  sont positifs.
- **Sous-tâche 6 [20 points]**: Pas de contrainte supplémentaire.

## Exemples

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

## Explication

Dans le **premier exemple**, la situation correspond à l'image ci-dessus. Le revenu optimal est obtenu avec  $x = 5$ .

Dans le **deuxième exemple**, le revenu optimal peut être obtenu en choisissant  $x = 5$ . Par conséquent, le revenu total est  $(5 + 5 + -2 + 5 + -5) - 1 \cdot 5 = 3$ .

Dans le **troisième exemple**, le revenu optimal peut être obtenu en choisissant n'importe quel  $x \geq 10^{12}$ .