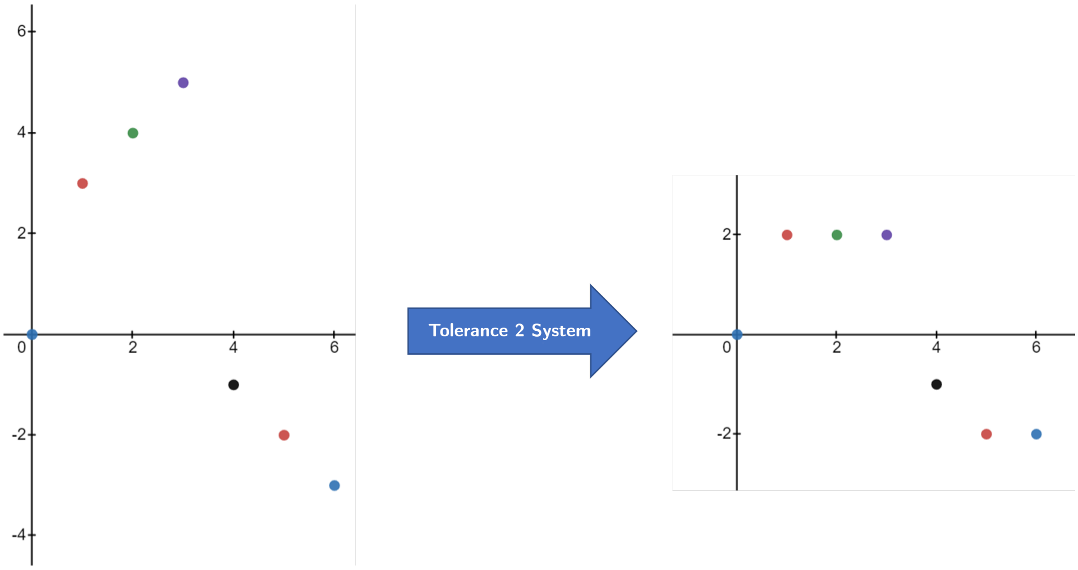
Plane Turbulences (`turbulences`)

Brianair, een gedegen vliegtuigmaatschappij, is onderzoek aan het doen naar turbulentie. Ze proberen het optimale stabilisatiesysteem te vinden voor hun vliegtuigen. Deze zal de hele vlucht actief zijn, met uitzondering van opstijgen en landen, dus het deel van de vlucht waar het vliegtuig *in een rechte lijn* hoort te vliegen.

Een stabilisatiesysteem met een speling x zorgt ervoor dat het vliegtuig niet afwijkt van hun gewenste hoogte (die die zou vliegen als het in een rechte lijn vliegt op constante hoogte) met een absoluut verschil groter dan x. Het is mogelijk om van te voren de hoogte van het vliegtuig voor elke minuut van de reis te weten als die geen stabilisatiesysteem heeft. Je krijgt al deze verwachte hoogteverschillen $A_0, \ldots, A_{N-1}$ voor de lengte van de vlucht in chronologische volgorde.

Het volgende voorbeeld toont hoe een stabilisatiesysteem met speling 2 een vlucht met verwachte afwijkingen $A_0 = 0, A_1 = 3, A_2 = 4, A_3 = 5, A_4 = -1, A_5 = -2, A_6 = -3$ wordt omgezet in een vlucht met werkelijke afwijkingen $B_0 = 0, B_1 = 2, B_2 = 2, B_3 = 2, B_4 = -1, B_5 = -2, B_6 = -2$.



Hoogtes voor en na het toepassen van een stabilisatiesysteem met speling 2.

Brianair weet dat klanten houden van hoog-vliegende vluchten, dus de klanttevredenheid (oftewel hun opbrengst door het implementeren van het systeem) na het vliegen op een vliegtuig met een stabilisatiesysteem met speling x is gelijk aan $\sum_{i=1}^{N} B_i$, waarbij B_i de gestabiliseerde hoogte van moment i is. Dus $B_i = \text{sign}(A_i) \cdot \min(|A_i|, x)$.

Maar de kosten om regelgevers te overtuigen een systeem met speling x toe te staan is Kx, waarbij K een niet-negatieve constante. De vliegtuigmaatschappij wil een zo groot mogelijke winst van de vlucht, oftewel $\left(\sum_{i=1}^{N} B_i\right) - Kx$.

Gegeven K en $A_1, \ldots, A_n$, kan je de maximale winst vinden die kan worden behaald door het kiezen van de optimale speling $x \geq 0$?

Implementatie

Je moet één `.cpp`-bestand inleveren.

☞ In de bijlagen van deze opgave vind je een sjabloon `turbulences.cpp` met een voorbeeld-implementatie.

Je moet de volgende functie implementeren:

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

- Integer N die staat voor de lengte van de vlucht.
- Integer K die staat voor de kosten-coëfficiënt.
- De array A, genummerd van 0 to $N-1$, bevat de getallen $A_0, A_1, \ldots, A_{N-1}$, waarbij A_i de verwachte hoogte is op moment i.
- De functie moet de maximaal haalbare winst teruggeven.

De grader roept de functie `revenue` aan en zal de teruggegeven waarde schrijven naar het uitvoerbestand.

Voorbeeld Grader

De map van de opdracht bevat een voorbeeldgrader, een versimpelde versie van de jury's grader, die je kan gebruiken om je oplossing lokaal te testen. De voorbeeldgrader leest de invoer van `stdin`, roept de functies aan die je moet implementeren en schrijft vervolgens de uitvoer naar `stdout`.

De invoer bestaat uit 2 regels met:

- Regel 1: de integers N and K.
- Regel 2: de getallen A_i, gescheiden door spaties.

De uitvoer bestaat uit één regel die de waarde bevat die door de functie `revenue` wordt teruggegeven.

Randvoorwaarden

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

Scoring

Je programma zal worden getest op een set van testgevallen die per deelopgaven zijn gegroepeerd. Om de score van een deelopgave te krijgen moet je alle testgevallen daarvan oplossen.

- **Subtask 1** [**0 punten**]: Voorbeeld testgevallen.
- **Subtask 2** [**15 punten**]: $N = 1$.
- **Subtask 3** [**30 punten**]: $N \leq 10^2$, $K \leq 10^2$, $-10^2 \leq A_i \leq 10^2$ voor elke $i = 0, \ldots, N-1$.
- **Subtask 4** [**17 punten**]: Alle A_i zijn gelijk.
- **Subtask 5** [**18 punten**]: Alle A_i zijn niet-negatief.
- **Subtask 6** [**20 punten**]: Geen extra randvoorwaarden.

Voorbeelden

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

Uitleg

In het **eerste voorbeeld** is de situatie zoals omschreven door het plaatje hierboven. De optimale winst kan worden behaald met $x = 5$.

In het **tweede voorbeeld** kan de optimale winst worden behaald met $x = 5$. Dus de totale winst is $(5 + 5 + -2 + 5 + -5) - 1 \cdot 5 = 3$.

In het **derde voorbeeld**, de optimale winst kan worden behaald door elke $x \geq 10^{12}$.