



Tömb csökkentése (subarrayd-pp)

Andrej és Alex egy N elemből álló V tömböt használnak egy játékhoz. Andrej legfeljebb K műveletet hajthat végre a tömbön. Minden művelet során bármelyik elemet 1-gyel csökkentheti. A műveletek elvégzése után Andrejnek S dollárt kell fizetnie Alexnek, ahol S a módosított tömb maximális összegű résztömbjének összege. (A tömb egy résztömbjét úgy kapjuk, hogy egy vagy több szomszédos elemet kiválasztunk.) Mivel Andrej a lehető legkevesebbet akarja fizetni, a műveleteit úgy választja meg, hogy S értéke a lehető legkisebb legyen. Adott N , K és a V tömb, mennyi a legkevesebb pénz, amit Andrejnek fizetnie kell Alexnek?

 Az értékelő rendszerből letölthető csatolmányok közt találhatsz `subarrayd.*` nevű fájlokat, melyek a bemeneti adatok beolvasását valósítják meg az egyes programnyelveken. A megoldásodat ezekből a hiányos minta implementációkból kiindulva is elkészítheted.

Bemenet

Az első sor két egész számot tartalmaz, N -et és K -t.

A második sor N darab egész számot tartalmaz: $V_0, V_1, \dots, V_{N-1}$.

Kimenet

Írj ki egy egész számot, ami megadja az S értékét.

Korlátok

- $1 \leq N \leq 200\,000$.
- $0 \leq K \leq 1\,000\,000\,000$.
- $-1\,000\,000\,000 \leq V_i \leq 1\,000\,000\,000$ minden $i = 0 \dots N - 1$ esetén.

Pontozás

A megoldásodat sok különböző tesztesetre lefuttatjuk. A tesztesetek részfeladatokba vannak csoportosítva. Egy-egy részfeladatot akkor tekintünk megoldottnak, ha volt legalább egy olyan beadásod, amely az adott részfeladat minden tesztesetére helyes megoldást adott. A feladat összpontszámát a megoldott részfeladatokra kapott pontszámok összege adja.

– **0. Részfeladat** (0 pont) Példák.



– **1. Részfeladat** (5 pont) $K = 0$.



– **2. Részfeladat** (15 pont) $K = 1$.



– **3. Részfeladat** (5 pont) $0 \leq V_i \leq 1\,000\,000\,000$ minden $i = 0 \dots N - 1$ esetén.



– **4. Részfeladat** (15 pont) $-50 \leq V_i \leq 50$ minden $i = 0 \dots N - 1$ esetén, $K \leq 50$.



– 5. Részfeladat (60 pont) Nincsenek további megkötések.



Példák

input	output
5 2 -1 2 3 -1 4	6
6 7 -2 2 2 3 0 4	4

Magyarázat

Az **első példában** az ötödik elemet kétszer csökkentheted.

A **második példában** a negyedik elemet hétszer csökkentheted.