

Fűnyíró (Lawnmower)

A Poenari erődben átélt kalandok után Vlad hazatér, és igazi románként az első gondolata az, hogy meg kell etetnie a lovát. A ló nem túl válogatós, ha ételről van szó, így Vlad a pázsitját használja elsődleges táplálékforrásként számára.

Ehhez a feladathoz Vladnak van egy c kapacitású fűnyírója. Úgy döntött, hogy a gyepet n sávra osztja, amelyek 0-tól $n - 1$ -ig vannak számozva, és amelyeket ebben a sorrendben kell lenyírnia. Minden egyes i sávban $v[i]$ mennyiségű vágatlan fű van, és ismeretlen okok miatt Vladnak $a[i]$ másodpercbe telik, amíg a fűnyírót végigtolja az adott sávon.

Néhány sáv lenyírása után a fűnyíró elérheti a teljes kapacitását. Esetben abbahagyja a fűnyírást, így a sávban marad valamennyi vágatlan fű. Minden alkalommal, amikor ez megtörténik, a gyűjtőtartályt ki kell üríteni, ami b másodpercet vesz igénybe, és csak a sáv végén történhet meg. Ha a gyűjtőtartály megtelik, miközben Vlad az i sávban halad, akkor a sáv végéig kell tolnia a fűnyírót, ki kell ürítenie a tartályt, majd még legalább egyszer (lehet, hogy többször is) át kell mennie a sávon, hogy levágja a megmaradt fűvet. Például ha az i -edik sávon 3-szor kell végigmennie, hogy lenyírja az összes fűvet, akkor az $a[i] + b + a[i] + b + a[i]$ másodpercbe telik, amíg az i sávban lévő összes fűtől megszabadul.

Miután lenyírta az összes fűvet, a fűnyírót ki kell üríteni.

Vlad hosszas gondolkodás és panaszkodás után, hogy túl sokáig tart neki befejezni a fűnyírást, arra a következtetésre jutott, hogy néha talán időhatékonyabb lenne kiüríteni a gyűjtőtartályt, még mielőtt az elérné a teljes kapacitását, de nem biztos benne, hogy mi a legjobb stratégia, amit alkalmazhat. Ezért az a te segítségedet kéri.

Ha adott az egyes sávokon lévő fű mennyisége és az egyes sávokon a fűnyíró tolására fordított másodpercek száma, a tartály kapacitása és az ürítéséhez szükséges idő, akkor találd meg a legjobb módszert, amellyel Vlad a lehető legkevesebb idő alatt befejezheti a fűnyírást.

Implementációs részletek

A következő függvényt kell implementálnod

```
long long mow(int n, int c, int b, std::vector<int> &a, std::vector<int> &v);
```

- n : a sávok száma a pázsiton
- c : a gyűjtőtartály kapacitása
- b : a kiürítéshez szükséges másodpercek száma
- a : egy n hosszú tömb, amely leírja, hogy az egyes sávokon mennyi idő alatt lehet végigmenni

- v : egy n hosszú tömb, amely megadja, hogy az egyes sávokban mennyi lenyírandó fű van
- Ez a függvény egy egész számot kell visszaadjon, a pázsit lenyírásához szükséges minimális időt
- Ez a függvény tesztelésenként egyszer hívódik meg

Korlátok

- $1 \leq n \leq 200\,000$
- $1 \leq a[i] \leq 10^9$ (minden i -re, melyre $0 \leq i < n$)
- $1 \leq v[i] \leq 10^9$ (minden i -re, melyre $0 \leq i < n$)
- $1 \leq b \leq 10^9$
- $1 \leq c \leq 10^9$
- Garantált, hogy a helyes válasz legfeljebb 10^{18}

Részfeladatok

1. (9 pont) Minden megadott érték ($n, b, c, a[i]$ és $v[i]$) legfeljebb 200
2. (16 pont) $n, c \leq 5000$ és $v[i] \leq 5000$ minden $0 \leq i < n$ esetén
3. (36 pont) $c \leq 200\,000$
4. (17 pont) $a[0] = a[1] = \dots = a[n-1]$
5. (22 pont) Nincsenek további megkötések.

Példák

1. példa

Legyen a következő hívás:

```
mow(3, 5, 2, {2, 10, 3}, {2, 4, 6})
```

Ebben a példában 3 sáv van, a gyűjtőtartály kapacitása 5, és 2 másodpercbe telik kiüríteni.

Ennél a példánál Vlad az első sávot 2 másodperc alatt nyírja le. A fű mennyisége a fűnyíróban 2 lesz. Ezután kiüríti a fűnyíró 2 másodperc alatt. Az első sávra így 4 másodpercet fordít.

Ezután a második sávon fog áthaladni. 4 egységnyi fűvet fog lenyírni. A második sáv befejezése után úgy dönt, hogy nem üríti ki a tartályt. A második sávra fordított idő 10 másodperc.

A harmadik sávban elkezd a fűnyírást. Egy egységnyi fű után a fűnyírója megtelik, így a sáv végéig kell mennie, kiüríteni a fűnyíró, majd újra elkezdni lenyírni a harmadik sávot. Ne feledd, hogy az egész udvar lekasálása után a fűnyíró ki kell üríteni. A harmadik sávra fordított idő így $3 + 2 + 3 + 2 = 10$ másodperc.

Összesen tehát $4 + 10 + 10 = 24$ másodpercet tölt el. Bizonyítható, hogy ez az optimális stratégia, amelyet Vlad a fűnyírásra használ.

2. Példa

Legyen a következő hívás:

```
mow(4, 10, 4, {1, 2, 1, 4}, {3, 2, 6, 7})
```

Ebben a példában 4 sáv van, a gyűjtőtartály kapacitása 10, és 4 másodpercbe telik kiüríteni.

Az optimális stratégia az, hogy csak az első 3 sávon megyünk át, ami során a tartály megtelik, és a fűmennyiségek vektora $[0, 0, 1, 7]$ lesz. Ezután a tartályt ki kell üríteni, majd az utolsó 2 sávot le kell nyírni, és a végén a tartályt ismét ki kell üríteni.

Tehát összesen $a[0] + a[1] + a[2] + b + a[2] + a[3] + b = 17$ másodperc alatt tudja lenyírni a fűvet.

Minta értékelő

A minta értékelő a bemenetet a következő formátumban olvassa:

- 1. sor: n, c, b
- 2. sor: $a[0] \ a[1] \ \dots \ a[n - 1]$
- 3. sor: $v[0] \ v[1] \ \dots \ v[n - 1]$

és kiírja a `mow` függvény eredményét, amelyet a megfelelő beolvasott paraméterekkel hív meg.