

## Kétféle répa

Pom-pom nyuszi répát szed. Sorban húzza ki a földből a répákat. Minden kihúzott répának ismeri a súlyát, valamint azt, hogy fehérrépa vagy sárgarépa-e. Ha üres a kosara, vagy egy olyan répát húz ki, aminél nagyobb, vagy ugyanakkorát már látott **az adott típusból**, akkor beleteszi a répát a kosárba. Ha azonban a kihúzott répa az adott típusból az eddigi legnagyobb, akkor örömeiben ugrálni kezd és az összes ugyanolyan típusú répa kiesik a kosarából. Ezután beleteszi a répát a kosárba és folytatja a répaszedést.

Írj programot, ami kiszámítja, hogy a répaszedés végén hány darab répa lesz a kosásban, és mekkora a kosárban lévő répák összszúlya.

### Bemenet

A standard bemenet első sorában a répák  $N$  száma található.

A következő  $N$  sor mindegyike két egész számot tartalmaz, az  $i$ -edik sor az  $i$ -edik répa  $S_i$  súlyát és  $T_i$  típusát ( $0 =$  fehérrépa,  $1 =$  sárgarépa).

### Kimenet

A standard kimenetre egy sort kell írni két egész értékkel: az első a kosárban lévő répák száma, a második pedig a az összszúlya a répaszedés végén.

### Példa

Bemenet	Kimenet
6	4 16
3 0	
5 1	
2 0	
4 1	
6 0	
1 1	

A répák súlya és típusa rendre:  $(3, F)$ ,  $(5, S)$ ,  $(2, F)$ ,  $(4, S)$ ,  $(6, F)$ ,  $(1, S)$ , ahol  $F$  a fehérrépa és  $S$  a sárgarépa.

- Kezdetben a kosár üres. Az első répa  $(3, F)$ , ezt beteszi a kosárba.
- A második répa  $(5, S)$ , ez az első sárgarépa, beteszi a kosárba.
- A harmadik répa  $(2, F)$ , ez kisebb a legnagyobb fehérrepánál  $(3)$ , beteszi a kosárba.
- A negyedik répa  $(4, S)$ , ez kisebb a legnagyobb sárgarépánál  $(5)$ , beteszi a kosárba.
- Az ötödik répa  $(6, F)$ , ez eddig a legnagyobb fehérrépa, ezért ugrál és a korábbi fehérrepák kiesnek. A  $(6, F)$  répát beteszi a kosárba.
- A hatodik répa  $(1, S)$ , ez kisebb a legnagyobb sárgarépánál  $(5)$ , beteszi a kosárba.

A végén 4 darab répa van a kosárban, amik összszúlya 16.

Bemenet

5  
10 0  
3 0  
5 0  
2 1  
1 1

Kimenet

5 21

**Korlátok**

$$1 \leq N \leq 200\,000$$

$$1 \leq S_i \leq 10^9 \text{ minden } i = 1 \dots N\text{-re}$$

$$T_i \in \{0, 1\} \text{ minden } i = 1 \dots N\text{-re}$$

**Időlimit:** 1.0 mp.**Memórialimit:** 256 MB**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.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	minden répa ugyanolyan típusú (mind fehérrépa vagy mind sárgarépa)	20
2	$N \leq 1000$	21
3	minden $S_i$ érték egyedi	22
4	nincsenek további megkötések	17

## Derékszögű háromszög

Pütagorasz birodalmának főépítésze egy monumentális templomtetőt tervez. A tetőszerkezet stabilitását a szent átfogók, az úgynevezett *hipotenuzák* biztosítják.

A tetőelemhez a főépítész már legyártatott egy  $C$  hosszúságú főgerendát. A raktárban azonban hatalmas káosz uralkodik:  $N$  darab támasztógerenda hever egymás után egy sorban. Az  $i$ -edik gerenda hossza  $A_i$ .

A feladatod segédkezni az építkezésen: meg kell számolnod a gerendák sorában azokat a párokat, amelyek a  $C$  hosszúságú főgerendával, mint átfogóval együtt egy tökéletes derékszögű háromszöget alkotnak. Formálisan azokat az  $(i, j)$  indexpárokat kell megszámolnod, amikre  $i < j$  teljesül, és az  $A_i, A_j, C$  oldalú háromszög derékszögű háromszög, aminek  $C$  az átfogója.

Hány ilyen szabályos támaszték-párt lehetséges kiválasztani a raktárkészletből?

### Bemenet

A standard bemenet első sorában a gerendák  $N$  száma és a főgerenda  $C$  hossza található.

A második sor  $N$  pozitív egész számot tartalmaz, rendre a raktárban lévő gerendák  $A_i$  hosszait.

### Kimenet

A standard kimenetre egyetlen egész számot kell írni: a feltételeknek megfelelő gerendapárok számát.

### Példa

Bemenet	Kimenet
5 5	1
3 4 1 2 6	

Ebben a példában  $N = 5$ ,  $C = 5$  és a gerendák hossza rendre  $3, 4, 1, 2, 6$ . Az egyetlen megfelelő pár az első és a második gerenda, mivel ezek az  $5$  hosszúságú gerendával derékszögű háromszöget alkotnak:  $3^2 + 4^2 = 9 + 16 = 25 = 5^2$ . Tehát a válasz  $1$ .

Bemenet	Kimenet
8 13	4
5 12 3 4 5 12 7 8	

Most  $N = 8$ ,  $C = 13$  és a gerendák hossza rendre  $5, 12, 3, 4, 5, 12, 7, 8$ . Itt a megfelelő indexpárok:  $(1, 2), (1, 6), (2, 5), (5, 6)$ . Tehát a válasz  $4$ .

### Korlátok

$$1 \leq N \leq 200\,000$$

$$1 \leq C \leq 10^9$$

$$1 \leq A_i \leq 10^9 \text{ minden } i = 1 \dots N\text{-re}$$

**Időlimit:** 0.6 mp.

**Memórialimit:** 256 MB

## Pontozás

A megoldásokat 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.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	$N \leq 100$ és $C \leq 100$	21
2	minden $A_i$ érték egyedi	24
3	nincsenek további megkötések	35

## Sorkizárt szöveg

Modern szövegszerkesztő programok fontos funkciója a sorkizárt tördelés. Ez egy olyan algoritmus, amely egy adott szélességű oszlopba tördeli a szöveget úgy, hogy a sorok a lehető *legszebben* töltsék ki a rendelkezésre álló helyet.

Adott egy  $N$  darab szóból álló szöveg. Mindegyik szónak ismerjük az alapértelmezett szélességét: rendre  $w_1, w_2, \dots, w_N$  képpont. A szöveget sorokra kell tördelni a szavak sorrendjének megőrzésével.

A képernyőn minden sor azonos,  $L$  képpont szélességű. Egy sorba tetszőleges számú és összszélességű szó kerülhet. Speciálisan a szavak összszélessége lehet  $L$ -nél nagyobb. A sorkizárt igazításnál a szavak közötti távolságot és a karakterek szélességét módosítjuk úgy, hogy a sor vége éppen elérje a margót. Egy adott sor **csúnyasági pontszáma** a sorban lévő szavak összszélessége és a sorhossz különbségének a négyzete. Formálisan, ha egy sorban a szavak összszélessége  $S$ , akkor a sor csúnyasági pontszáma  $(L - S)^2$ . A teljes szöveg csúnyasági pontszáma a sorok csúnyasági pontszámának összege.

A cél a teljes szöveg tördelése úgy, hogy a csúnyasági pontszám minimális legyen. Írj programot, ami kiszámítja, mi az elérhető legkisebb csúnyasági pontszám és hogy legkevesebb hány sorba szükséges tördelni a szöveget ennek eléréséhez.

### Bemenet

A standard bemenet első sorában szavak  $N$  száma és a sorok  $L$  szélessége található.

A második sorban  $N$  darab egész szám van, a szavak alapértelmezett  $w_i$  szélességei a szöveg szerinti sorrendjükben.

### Kimenet

A standard kimenetre egy sort kell írni két egész értékkel: az első szám megadja, hogy mennyi a legkisebb elérhető csúnyasági pontszám, a második pedig, hogy legkevesebb hány sorra van szükség az eléréséhez.

### Példa

Bemenet	Kimenet
5 8	5 2
3 3 3 3 7	

Az első három szó az első, az utolsó két szó a második sorba kerül.

Ekkor a csúnyasági pontszám:  $(8 - 9)^2 + (8 - 10)^2 = 5$ .

Bemenet	Kimenet
5 7	2 3
3 3 3 3 7	

### Korlátok

$$1 \leq N \leq 200\,000$$

$$1 \leq L \leq 100$$

$$1 \leq w_i \leq L \text{ minden } i = 1 \dots N\text{-re}$$

**Időlimit:** 1.5 mp.

**Memórialimit:** 256 MB

## Pontozás

A megoldásokat 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.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	$N = 2$	8
2	$W_1 + W_2 + \dots + W_N \leq 2L$	10
3	$N \leq 10\,000$	30
4	nincsenek további megkötések	32

## Festés

Patrik festő. Egy nap különleges megbízást kapott: az ügyfele pontosan olyanra szeretné lefestetni a kerítését, amilyen a szomszédja kerítése. Mindkét kerítést egy  $N$  sorból és  $M$  oszlopból álló ráccsal modellezzük. A szomszéd kerítésén minden mező háromféle lehet:

- festetlen ('.'),
- világos színű ('V'), vagy
- sötét színű ('S').

Az ügyfél kerítésén kezdetben minden mező festetlen ('.'). Munka közben Patrik minden lépésben kiválaszt egy sort vagy oszlopot az ügyfél kerítésén, és azt teljes egészében lefesti a két szín (világos vagy sötét) egyikével. Ha egyes mezők le voltak már festve korábban az adott sorban vagy oszlopban, azokat is átfesti az új színnel. Minden sort vagy oszlopot legfeljebb egyszer választhat ki Patrik a festés során.

Patrik tapasztalt szakember, így ránézésre tudja, hogy a módszerével le tudja másolni a szomszéd kerítését. Írj programot, ami megad egy tetszőleges lépéssorozatot, amivel eléri a célját.

### Bemenet

A standard bemenet első sorában a rács sorainak  $N$  és oszlopainak  $M$  száma található.

A következő  $N$  sor mindegyike egy  $M$  hosszúságú  $T_i$  karakterláncot tartalmaz, a szomszéd kerítésének leírását '.', 'V' és 'S' karakterekkel.

### Kimenet

A standard kimenet első sorába a lépések  $K$  száma kerüljön.

A következő  $K$  sor mindegyike egy lépést ír le egy egész szám és egy karakter segítségével:

- Az egész szám egy sor vagy oszlop azonosítója. A sorokat rendre az  $1, 2, \dots, N$ , az oszlopokat az  $N + 1, \dots, N + M$  számokkal azonosítjuk.
- A karakter a festéshez használt szín azonosítója ('V' vagy 'S').

Több lehetséges megoldás esetén bármelyik megadható.

### Példa

Bemenet	Kimenet
4 4	2
..S.	2 V
VVSV	7 S
..S.	
..S.	

Az első példában Patrik először a második sort (azonosító: 2) festi világosra, majd a harmadik oszlopot (azonosító: 7) sötétre.

Bemenet	Kimenet
5 5	5
VVVV	8 S
..S.S	3 V
VVVVS	10 S
VVVVV	4 V
..S.S	1 V

### Korlátok

$$1 \leq N, M \leq 2000$$

Minden  $i = 1 \dots N$ -re  $T_i$  hossza  $M$  és csak a '.', 'V' és 'S' karaktereket tartalmazza.

Legalább egy mező színes.

Létezik legalább egy lépéssorozat a kerítés megfelelő lefestésére.

**Időlimit:** 1.5 mp.

**Memórialimit:** 256 MB

### 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.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	$N = 1$	15
2	$N, M \leq 4$	21
3	$N, M \leq 10$	22
4	nincsenek további megkötések	22

## Három csoport

Egy zenetagozatos iskolába összesen  $N$  tanuló jár, akiket 1-től  $N$ -ig sorszámozunk. Minden tanuló énekkaros, és már nagyon készülnek a karácsonyi műsorra. Összesen három előadást szerveznek, amihez három csoportba (1-es, 2-es, 3-as) kell beosztani a tanulókat úgy, hogy minden tanuló pontosan egy csoportnak legyen a tagja.

A műsort az alábbiak szerint tervezik:

- Az 1-es csoport előadását a 2-es csoport fogja meghallgatni (közben a 3-as csoport felkészül).
- A 2-es csoport előadását a 3-as csoport fogja meghallgatni (közben az 1-es csoport felkészül).
- A 3-as csoport előadását az 1-es csoport fogja meghallgatni (közben a 2-es csoport felkészül).

Minden tanulónak van egy  $A_i$  elégedettségi küszöbértéke. Ez azt jelenti, hogy az  $i$  sorszámú tanuló akkor lesz elégedett a saját csoportja előadásával, ha azt legalább  $A_i$  tanuló hallgatja meg. Például, ha az  $i$  sorszámú tanuló az 1-es csoportba kerül, akkor ő pontosan akkor lesz elégedett, ha a 2-es csoportnak legalább  $A_i$  tagja van.

Írj programot, ami a küszöbértékek ismeretében meghatározza, hogy be lehet-e osztani a tanulókat három csoportba úgy, hogy végül **minden** tanuló elégedett lesz. Ha létezik ilyen beosztás, akkor meg is kell adni egyet.

### Bemenet

A standard bemenet első sorában a tanulók  $N$  száma található.

A második sor  $N$  pozitív egész számot tartalmaz, a tanulók  $A_i$  elégedettségi küszöbértékeit.

### Kimenet

A standard kimenet első sorába az IGEN szöveg kerüljön, ha létezik megfelelő beosztás. A második sor  $N$  darab egész értéket tartalmazzon, amik egy szabályos beosztást írnak le: az  $i$ -edik szám annak a csoportnak az azonosítója legyen, amibe az  $i$ -edik tanuló kerül.

Ha nem létezik megfelelő beosztás, akkor egyetlen sorba a NEM szöveget kell kiírni.

### Példa

Bemenet	Kimenet
10	IGEN
1 3 1 3 3 2 4 1 5 2	3 3 1 3 3 3 2 1 2 1

Az első példában:

- a 2-es csoportnak 2 tagja van és az 1-es csoport minden tagjának küszöbértéke legfeljebb 2;
- a 3-as csoportnak 5 tagja van és a 2-es csoport mindkét tagjának küszöbértéke legfeljebb 5;
- az 1-es csoportnak 3 tagja van és a 3-es csoport minden tagjának küszöbértéke legfeljebb 3.

Bemenet	Kimenet
3	NEM
1 2 2	

## Korlátok

$$3 \leq N \leq 500\,000$$

$$1 \leq A_i \leq N \text{ minden } i = 1 \dots N\text{-re}$$

**Időlimit:** 2.5 mp.

**Memórialimit:** 256 MB

## 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.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	minden $A_i$ érték azonos ( $A_1 = A_2 = \dots = A_N$ )	11
2	$N \leq 10$	15
3	$A_i \leq 3$	9
4	$N \leq 3000$	19
5	nincsenek további megkötések	26