Amőba

Az amőba játékban két játékos felváltva helyez el `O` és `X` jeleket egy 3×3-as játéktáblán. Kezdetben a tábla mezői üresek, és az első játékos mindig az `O` jelet helyezi le.

Egy játékban eddig `K` lépés történt. Írj programot, ami kirajzolja a játék aktuális állását!

Bemenet

A `standard bemenet` első sorában a megtett lépések `K` száma van.

A következő `K` sor mindegyike két pozitív egészet tartalmaz, egy lépést leíró x_i és y_i számokat: a soron következő játékos az x_i-edik sor y_i-edik cellájába helyezett jelet.

Kimenet

A `standard kimenetre` a játéktábla állapotát kell kirajzolni `K` lépés megtétele után az alábbi példában látható formátumban, a `+`, `-`, `|`, `O`, `X` és szóköz karakterek felhasználásával.

Példa

Bemenet

```
3
1 2
2 1
3 3
```

Kimenet

```
+-+-+-+
| |O| |
+-+-+-+
|X| | |
+-+-+-+
| | |O|
+-+-+-+
```

Magyarázat: az első játékos először az első sor második cellájába helyezett `O` jelet, erre a második játékos a második sor első cellájába tett egy `X`-et, végül az első játékos a harmadik sor harmadik cellájába rakott `O`-t.

Korlátok

$0 \leq K \leq 5$

$1 \leq x_i, y_i \leq 3$ minden $i = 1 \ldots K$-ra

az (x_i, y_i) párok mind különbözőek

Időlimit: 1.0 s

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	K = 0	20
2	K = 1	30
3	nincsenek további megkötések	50

Legjobb edzéssorozat

Peti egyik újévi fogadalma, hogy minden egyes nap futni fog járni. Szeretné is azonnal betervezni a futóedzéseit a következő N napra. Peti nagyon elfoglalt, ezért az i-edik napon csak legfeljebb A_i percet tud a futásra szánni. Azonban szeretné úgy megtervezni az edzéseit, hogy közben érezhetően fejlődjön is és megmaradjon a lelkesedése. Ezért úgy döntött, hogy ha egy tetszőleges napon x percet szán futásra, akkor az azt követő napon is legalább x percet fog futni.

Írj programot, ami kiszámítja, hogy legfeljebb hány percet tud Peti futásra fordítani **összesen** a következő N nap során, ha minden nap futni fog, és minden nap legalább annyi ideig fog futni, mint az azt megelőző napon!

Bemenet

A `standard bemenet` első sorában a futással töltött napok N száma található.

A következő sor N darab pozitív egészet tartalmaz, az egyes napokon futásra fordítható percek maximális A_i számát.

Kimenet

A `standard kimenetre` egy sort kell írni egyetlen számmal, a futással töltött percet összegének legnagyobb értékét a feltételek betartása mellett.

Példa

Bemenet	Kimenet
10 5 6 7 8 9 3 2 7 8 9	38

Bemenet	Kimenet
4 1 2 2 3	8

Magyarázat: az első esetben 2, 2, 2, 2, 2, 2, 2, 7, 8, 9 lesz a futással töltött percek száma az egyes napokon. A második esetben rendre 1, 2, 2, 3 percet fog futni Peti az egyes napokon.

Korlátok

$1 \le N \le 300\,000$

$1 \le A_i \le 10\,000$ minden $i = 1 \ldots N$-re

Időlimit: 1.0 s

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	$N \leq 1000$ és $A_i \leq A_{i+1}$ minden $i = 1 \ldots N - 1$-re	25
2	$N \leq 1000$	25
3	nincsenek további megkötések	50

Kártyagyűjtemény

Krisztián nagy rajongója a gyűjtögetős kártyajátékoknak. Most épp a Bajnokok Ligájában játszó focisták gyűjthető kártyáit igyekszik megszerezni, és már csak `M` játékos kártyája hiányzik a kollekciójából. Az egyszerűség kedvéért számozzuk a hiányzó kártyákat `1`-től `M`-ig.

Hogy megszerezze a hiányzó lapokat, Krisztián úgy döntött, hogy vásárol `N` csomag bontatlan kártyacsomagot. Minden kártyacsomag sok játékos kártyáját tartalmazza, de Krisztiánt ezek közül csak a hiányzó játékosok kártyái érdeklik.

Írj programot, ami eldönti, hogy az újonnan vásárolt csomagok tartalmával teljessé teheti-e Krisztián a gyűjteményét!

Bemenet

A `standard bemenet` első sora két egész számot tartalmaz, a vásárolt kártyacsomagok `N`, és a hiányzó kártyák `M` számát.

A következő `N` sor mindegyike először egy k_i egészet tartalmaz, azoknak a kártyáknak a számát az `i`-edik csomagban, amik kezdetben hiányoztak Krisztián gyűjteményéből. Ezt k_i darab egész p_j érték követi, ezeknek a kártyáknak a sorszámai, tetszőleges sorrendben.

Kimenet

A `standard kimenet`re egy sort kell írni, melynek tartalma `IGEN`, ha minden hiányzó kártyát megszerzett Krisztián, vagy `NEM`, ha még nem teljes a gyűjteménye.

Példa

Bemenet	Kimenet
2 4 3 2 1 4 3 2 4 1	NEM

Bemenet	Kimenet
3 6 3 1 6 3 6 1 2 3 4 5 6 2 5 1	IGEN

Bemenet	Kimenet
0 4	NEM

Bemenet	Kimenet
3 0 0 0 0	IGEN

Korlátok

$0 \leq \mathtt{N}, \mathtt{M} \leq 500\,000$

$0 \leq \mathtt{N} \cdot \mathtt{M} \leq 500\,000$

$0 \leq k_i \leq M$ minden $i = 1 \ldots N$-re

$1 \leq p_j \leq M$ minden $j = 1 \ldots k_i$-re

Időlimit: 1.5 s

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	$N = 1$, $1 \leq M \leq 100$, a p_j értékek növekvő sorrendben vannak	10
2	$M = 0$ és $N \leq 100$	10
3	$N = 0$ és $M \leq 100$	10
4	$N, M \leq 500$	35
5	nincsenek további megkötések	35

Kétes dicsőség

Andris és Gábor szeretnek biliárdozni, és gyakran el is járnak játszani egymás ellen néhány partit. Gábor nagyon szeret dicsekedni vele, hogy milyen jó eredményei vannak Andrissal szemben.

Csakhogy ezeket az eredményeket egy elég egyedi módon számolja Gábor: mindig attól a mérkőzéstől kezdve számolja csak a partikat a legutolsó lejátszott meccsükig, amitől nézve a győzelmeik száma közti különbség a legnagyobb lesz az ő javára (a döntetleneket nem veszi figyelembe). Gábor legalább a legutolsó lejátszott partit mindenképp számításba veszi. Formálisan, ha a legutolsó `k` lejátszott mérkőzésen Gábornak g_k, Andrisnak pedig a_k győzelme van, akkor arra a `k`-ra számolja Gábor az eredményt, amire $g_k - a_k$ különbség a legnagyobb. Több ilyen `k` esetén a legkisebb `k`-val fog számolni.

Vegyünk az `N` legutóbbi mérkőzésüket, melyek közül mindegyik eredménye vagy Andris győzelme (`A`), vagy Gábor győzelme (`G`), vagy döntetlen (`D`). Írj programot, ami meghatározza a Gábor számára legkedvezőbb egymás elleni eredményüket minden egyes lejátszott mérkőzést követően!

Bemenet

A `standard bemenet` első sorában a lejátszott partik `N` száma található.

A következő sor egy `N` hosszú karaktersorozatot tartalmaz, mely a mérkőzések eredményeit írja le az `A`, `G` és `D` karakterekkel.

Kimenet

A `standard kimenetre` `N` sort kell írni, az `i`-edik sor a Gábor számára legkedvezőbb eredményt tartalmazza az `i`-edik lejátszott mérkőzést követően. Az eredményt `g - a` formában kell kiírni, ahol `g` Gábor győzelmeinek száma, `a` Andris győzelmeinek száma a Gábor által akkor figyelembe vett mérkőzéseken.

Példa

Bemenet	Kimenet
3 GGA	1-0 2-0 2-1

Bemenet	Kimenet
5 AGAAD	0-1 1-0 1-1 0-1 0-0

Magyarázat: a második példában

- az első meccs után a legkedvezőbb eredmény `0-1`, mivel legalább a legutolsó meccset figyelembe kell vennie Gábornak;
- a második meccs után csak a második meccset veszi figyelembe, így az eredmény `1 - 0`;
- a harmadik meccs után az utolsó két meccset veszi figyelembe, így az eredmény `1 - 1`;
- a negyedik meccs után a legkedvezőbb eredmény csak az utolsó vereséget figyelembe véve adódik, ami `0 - 1` (ugyanez a különbség adódik a három legutóbbi meccset figyelembe véve `1 - 2`

eredménnyel, de azonos eredménynél a legkevesebb lejátszott meccset kell tekinteni);

- az ötödik lejátszott meccs után az eredmény `0 − 0`, mivel csak az utolsó döntetlent fogja számításba venni Gábor.

Korlátok

$1 \leq N \leq 300\,000$

Időlimit: 1.5 s

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	$N = 1$	10
2	Gábor a legkedvezőbb eredményt mindig az összes addig lejátszott mérkőzést tekintve érte el	20
3	$N \leq 80$	20
4	$N \leq 1000$	15
5	nincsenek további megkötések	35

Kertvárosi parkolás

Zsolt egy kertvárosi utcában lakik, amin egy kétsávos út halad keresztül. Az utca mindkét oldalán `N` ház található. Bár a házak mindegyikéhez tartozik garázs, a lakók sokszor lusták és inkább az utcán, pontosabban az úttesten állnak meg az autóikkal. Egy ház előtt parkoló autó mindig az út egyik sávját foglalja el magának. Formálisan az úttestet egy `2` $\times$ `N`-es táblázatként képzelhetjük el, ahol a két sor a két sávnak felel meg, az `N` oszlop pedig az úttest házak előtti szakaszait jelképezi. Például a

```
...o.
.....
```

táblázattal ábrázolhatjuk azt a helyzetet, amikor `N = 5` ház van, és az első sáv negyedik háza előtt áll egy autó.

Sajnos a lakók sokszor figyelmetlenek, és úgy állnak meg, hogy az utcán nem lehet áthaladni autóval. Akkor mondjuk azt, hogy keresztül lehet haladni az utcán, ha az első oszlop egyik cellájából el lehet jutni az utolsó oszlop egyik cellájába csak jobbra, felfelé és lefelé lépve úgy, hogy nem lépünk parkoló autóra. Ha az első vagy az utolsó oszlopban nincs szabad cella, akkor sem lehetséges keresztül haladni. Átlósan nem lehet lépni.

A parkoló autók folyamatosan jönnek-mennek, így összesen `K` eseményt kell feldolgoznod: minden esemény során vagy egy új autó leparkol, vagy egy már parkoló autó elmegy, vagy Zsolt megkérdezi, hogy az adott pillanatban keresztül lehet-e haladni az utcán.

Írj programot, ami kezeli az eseményeket és válaszol Zsolt kérdéseire!

Bemenet

A `standard bemenet` első sorában a házak `N` száma és a lépések `K` száma található.

A következő két sor egy-egy `N` hosszú karaktersorozatot tartalmaz, az utca kezdeti állapotának leírását `'.'` és `'o'` karakterekkel, ahol előbbi szabad úttestet, míg utóbbi parkoló autót jelöl.

A következő `K` sor mindegyike egy esemény leírását tartalmazza:

- ha a sor első karaktere `'U'`, akkor a sor két további, szóközzel elválaztott `x` és `y` számot tartalmaz. Ennek jelentése, hogy ha az `x`-edik sáv `y`-adik háza előtt (az `(x, y)` cellában) parkolt autó, akkor az elhajtott, ha nem parkolt autó, akkor mostantól parkol egy.
- ha a sor egyetlen `'Q'` karaktert tartalmaz, akkor Zsolt szeretné tudni, hogy keresztül lehet-e haladni az utcán.

Kimenet

A `standard kimenetre` Zsolt minden kérdésére az `IGEN` vagy a `NEM` választ kell kiírni, pontosan akkor `IGEN`-t, ha keresztül lehet haladni az utcán a kérdés pillanatában.

Példa

Bemenet	Kimenet

<pre>5 5
...o.
.....
U 2 3
Q
U 1 3
U 2 3
Q</pre>
	<pre>NEM
	IGEN</pre>

Magyarázat: Először a második sáv harmadik pozíciójába parkol egy új autó, most az utca így néz ki:

```
...o.
..o..
```

Ezért most nem lehet keresztül haladni az utcán. A következő két változásnál egy új autó érkezik az első sáv harmadik pozíciójába, és elmegy a parkoló autó a második sáv harmadik pozíciójából. Így az utca most így néz ki:

```
..oo.
.....
```

Tehát most keresztül lehet haladni az utcán.

Bemenet	Kimenet

<pre>5 7
ooooo
.....
Q
U 1 1
U 1 2
Q
U 1 2
U 1 1
Q</pre>
	<pre>IGEN
	IGEN
	IGEN</pre>

Bemenet	Kimenet

<pre>3 4
...
...
U 1 1
U 1 2
U 1 3
Q</pre>
	<pre>IGEN</pre>

Korlátok

$2 \leq N \leq 200\,000$

$1 \leq K \leq 200\,000$

$1 \leq x \leq 2$ és $1 \leq y \leq N$ minden változásnál

Időlimit: 2.0 s

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	$N, K \leq 100$	25
2	$K = 1$ és ez az egyetlen esemény 'Q' típusú	5
3	a második sorban minden ház előtt kezdetben autó áll, és ezek soha nem mennek el	30
4	nincsenek további megkötések	40