

Feladat Aprogressive

Bemenet `aprogressive.in`
 Kimenet `aprogressive.out`

Adott egy n soros (1-től n -ig sorszámozva) és m oszlopos (1-től la m -ig sorszámozva) T mátrix, melynek elemei egész számok.

A T mátrix egy részmátrixát a következők határozzák meg: az (x_1, y_1) , a bal felső sarok sorszáma és oszlopszáma, az (x_2, y_2) a jobb alsó sarok sorszáma és oszlopszáma, ahol $1 \leq x_1 \leq x_2 \leq n$ és $1 \leq y_1 \leq y_2 \leq m$, valamint a mátrix összes (x, y) eleme, amelyekre érvényes az $x_1 \leq x \leq x_2$ és $y_1 \leq y \leq y_2$. Sajátos eset: egy részmátrix akkor azonos a T mátrixal, ha a bal felső sarka az $(1, 1)$ és a jobb alsó sarka az (n, m) .

Egy adott részmátrix minden sorának az elemeinek összeadásával számítjuk ki a soronkénti összegeket. A részmátrix soronkénti elemeinek összegéből alkotunk, egy S sorozatot, amelynek elemei a fent említett összegek. Egy adott részmátrix *aprogressive*, ha $x_1 < x_2$ és $y_1 < y_2$ és a sorösszegek S sorozata átrendeázhető úgy, hogy minden tagjával együtt egy számtani haladványt képezzenek, melynek rációja (az állandó különbség) egy **nem nulla** r szám.

Jelöljük $\mathbb{C}(R)$ -el annak az R részmátrixnak a tömörített formáját, amelynek a bal felső sarka az (x_1, y_1) és a jobb alsó sarka az (x_2, y_2) , és a következőképpen van meghatározva:

- ha $x_1 = x_2$ (egy soros részmátrix) vagy ha $y_1 = y_2$ (egy oszlopos részmátrix) a tömörített formája $\mathbb{C}(R) = (x_1, y_1, x_2, y_2, 0)$. Különben,
- ha R egy *aprogressive*, akkor a tömörített formája $\mathbb{C}(R) = (x_1, y_1, x_2, y_2, r)$. Különben,
- az R -et felosztjuk 4 részmátrixra: A, B, C, D , melyek elemei diszjunkt halmazokat alkotnak, a szomszédos ábra szerint, és ahol az A részmátrix bal felső sarka az (x_1, y_1) , és a jobb alsó sarka az $(\lceil \frac{x_1+x_2}{2} \rceil, \lceil \frac{y_1+y_2}{2} \rceil)$, és ahol az $[x]$ az x szám egész részét jelöli. Az R részmátrix tömörített formája rekurzívan van definiálva: $\mathbb{C}(R) = (\mathbb{C}(A), \mathbb{C}(B), \mathbb{C}(C), \mathbb{C}(D))$.

(x1,y1)		
A		B
C		D
		(x2,y2)

Például, a mellékelt ábrán lévő T mátrixot, amelyben az $n = 5, m = 5, x_1 = 1, y_1 = 1, x_2 = 5, y_2 = 5$ az alábbiakat követve tömörítjük.

Mivel se nem sormátrix, se nem oszlopmátrix, meghatározzuk a T mátrix soronkénti összegeinek S sorozatát: 9, 19, 14, 8, 10. Mivel nem egy *aprogressive* mátrix, fel lesz osztva az alábbi részmátrixokra:

- A – amelynek bal felső sarka és jobb alsó sarka: $(1, 1)(3, 3)$;
- B – amelynek bal felső sarka és jobb alsó sarka: $(1, 4)(3, 5)$;
- C – amelynek bal felső sarka és jobb alsó sarka: $(4, 1)(5, 3)$;
- D – amelynek bal felső sarka és jobb alsó sarka: $(4, 4)(5, 5)$;

2	1	1	3	2
3	2	5	4	5
1	4	2	1	6
2	2	1	2	1
1	3	1	2	3

Az A részmátrix soronkénti összegeinek az S sorozata: 4, 10, 7, amelynek ha átrendeázuk az elemeit, akkor egy számtani haladványt kapunk, az $r = 3$ rációval. Az A részmátrix tömörített formája az: $\mathbb{C}(A) = (1, 1, 3, 3, 3)$.

A B részmátrix soronkénti összegeinek az S sorozata: 5, 9, 7, amelynek ha átrendeázuk az elemeit, akkor egy számtani haladványt kapunk, az $r = 2$ rációval. A B részmátrix tömörített formája az: $\mathbb{C}(B) = (1, 4, 3, 5, 2)$.

A C részmátrix soronkénti összegeinek az S sorozata: 5, 5, amelynek ha átrendeázuk az elemeit, akkor egy számtani haladványt kapunk, az $r = 0$ rációval. Ennél a részmátrixnál a felosztás megismétlődik. A C részmátrix tömörített formája az: $\mathbb{C}(C) = ((4, 1, 4, 2, 0)(4, 3, 4, 3, 0)(5, 1, 5, 2, 0)(5, 3, 5, 3, 0))$.

A D részmátrix soronkénti összegeinek az S sorozata: 3, 5, amelynek ha átrendeázuk az elemeit, akkor egy számtani haladványt kapunk, az $r = 2$ rációval. A D részmátrix tömörített formája az: $\mathbb{C}(D) = (4, 4, 5, 5, 2)$.

Az így kapott T részmátrix tömörített formája az:

$\mathbb{C}(T) = ((1, 1, 3, 3, 3)(1, 4, 3, 5, 2)((4, 1, 4, 2, 0)(4, 3, 4, 3, 0)(5, 1, 5, 2, 0)(5, 3, 5, 3, 0))(4, 4, 5, 5, 2))$.

Követelmények

Ismerve a T mátrix méreteit és az elmeit, határozzuk meg:

1. A T mátrix azon sorainak az indexét (sorszámát), amelyeknek a sorösszege a legnagyobb.
2. A T mátrix azon sorainak az indexét, amelyeknek az elmeit ha átrendeázuk, akkor egy olyan számtani haladványt kapunk, amelynek a rációja egy nem nulla érték lesz.
3. A T mátrix tömörített formáját.

Bemeneti adatok

Az `aprogressive.in` bemeneti állomány első sora egy c természetes számot tartalmaz, a megoldandó követelmény számát. A második soron, két természetes szám található, az n és az m a követelménynek megfelelően. A következő n sor mindegyikében m darab egész szám található, a T mátrix elemei. Az azonos sorba írt értékek szóközzel vannak elválasztva.

Kimeneti adatok

Az `aprogressive.out` állományban vagy az 1-es követelmény megoldása (ha $c = 1$) lesz, vagy a 2-es követelmény megoldása (ha $c = 2$) lesz, vagy a 3-as követelmény megoldása (ha $c = 3$) lesz.

Az 1-es követelmény esetében a megoldás indexei soronként, **szigorúan növekvő sorrendben**, lesznek kiírva.

A 2-es követelmény esetében a megoldás indexei soronként, **szigorúan növekvő sorrendben**, lesznek kiírva, és abban az esetben, amikor nincsenek a követelménynek megfelelő sorok, a 0 lesz kiírva.

A 3-es követelmény esetében a megoldás a T mátrix tömörített formája lesz. Az eredményt az állomány egyetlen sorában, szóközők nélkül, vesszővel elválasztva, megfelelő kerek zárójelekkel kell írni.

Korlátok

- $c \in \{1, 2, 3\}$
- $1 \leq n, m \leq 1024$
- A T mátrix elemei a $[-10^9, 10^9]$ intervallumbeliek.

#	Pontszám	Korlátok
1	20	$c = 1$
2	25	$c = 2$
3	25	$c = 3$ și $n, m \leq 512$
4	30	$c = 3$ egyéb korlátozások nélkül

Példák

<code>aprogressive.in</code>	<code>aprogressive.out</code>	Magyarázat
<pre>1 3 4 6 3 7 4 3 1 4 2 2 6 4 8</pre>	<pre>1 3</pre>	$c = 1, n = 3, m = 4$. Az 1-es követelmény megoldása. Az 1-es soron az összeg egyenlő 20-al. A 2-es soron az összeg egyenlő 10-el. A 3-as soron az összeg egyenlő 20-al. Az 1-es és 3-as sorok indexei lesznek kiírva, ebben a sorrendben, mert ezeken a sorokon az összeg maximális.
<pre>2 3 4 6 3 7 4 3 1 4 2 2 6 4 8</pre>	<pre>2 3</pre>	$c = 2, n = 3, m = 4$. A 2-es követelmény megoldása. A 2-es sor elemeit lehet úgy átrendezni, hogy számtani haladványt alkossanak, $r = 1$-es rációval: 1, 2, 3, 4. A 3-as sor elemeit lehet úgy átrendezni, hogy számtani haladványt alkossanak, $r = 2$-es rációval: 2, 4, 6, 8. A 2-es és a 3-as sorok indexei lesznek kiírva, ebben a sorrendben.

Példák

aprogressive.in	aprogressive.out	Magyarázat
3	((1,1,3,3,3))	$c = 3, n = 5, m = 5$. A 3-as követelmény megoldása.
5 5	(1,4,3,5,2)	A megoldás egyetlen sorban jelenik meg a kimeneti állományban.
2 1 1 3 2	((4,1,4,2,0))	A megoldás lépéseit a felhívó szöveg ismerteti.
3 2 5 4 5	(4,3,4,3,0)	
1 4 2 1 6	(5,1,5,2,0)	
2 2 1 2 1	(5,3,5,3,0))	
1 3 1 2 3	(4,4,5,5,2))	

Feladat Opsir

Bemenet `opsir.in`
Kimenet `opsir.out`

Adott két karakterlánc, az S és a T , amelyek hossza n és m . A karakterláncok kizárólag az angol ábécé kisbetűiből állnak. A betűk sorszámozása 1-től kezdődik.

A T karakterláncon kétféle műveletet lehet végezni:

- 1 p : a p pozíción lévő betű törlődik;
- 2 $st\ dr$ (ahol $st \leq dr$) : a $[st, dr]$ intervallumban lévő pozícióknak megfelelő részkarakterlánc betűi növekvő sorrendbe (ábécé sorrendbe) lesznek rendezve;

ahol p , st és dr a T karakterlánc betűinek pozíciói.

Kezdetben, a T sorozat összes betűje szintelen. Egy 2-es típusú művelet csak akkor hajtható végre, ha a $[st, dr]$ intervallumban lévő pozícióknak megfelelő részkarakterlánc összes betűje szintelen. A rendezés után, a részkarakterlánc minden betűje színes lesz.

Követelmények

Minden adott S és T karakterlánc-pár esetén:

1. Írassuk ki azokat a különböző betűket, amelyek legalább az egyik karakterláncban előfordulnak, és mind-egyiknél annak a karakterláncnak a szimbólumát (az S vagy T betűket), amelyben leggyakrabban fordul elő. Ha ugyanannyiszor fordul elő, akkor a T szimbólum lesz kiírva.
2. Határozzunk meg 1-es és/vagy 2-es típusú művelet sorokat, amelyeket alkalmazva a T karakterláncra az átalakul S -re. Írassuk ki a DA szöveget, ha létezik ilyen művelet sor, vagy a NU szöveget, ha nem létezik.

Bemeneti adatok

Az `opsir.in` bemeneti állomány első sora egy c természetes számot tartalmaz, a megoldandó követelmény számát (1 vagy 2), minden karakterlánc-párra.

Az állomány második sora a k természetes számot tartalmazza, amely a tesztek számát jelenti. Az állomány következő soraiban minden tesztnek megfelelő három-három sorban a karakterlánc-pár adatai vannak, az alábbiak szerint:

az első sorban az n és az m , ebben a sorrendben, a felhívó szövegnek megfelelően, a második sorban az S karakterlánc, a harmadik sorban pedig a T karakterlánc.

Kimeneti adatok

A $c = 1$ esetén, az `opsir.out` kimeneti állomány tartalmazni fog, minden egyes teszt esetén, egy-egy nr természetes számot, amely azon különböző betűk számát jelöli, amely betűk megjelennek valamelyik karakterláncban, és a következő nr sorban egy-egy ilyen betű, valamint az S vagy a T , annak megfelelően, hogy melyik karakterláncban szerepel többször. A kisbetűk ábécé sorrendben lesznek kiírva. A $c = 2$ esetén, az `opsir.out` kimeneti állománynak k sora lesz, minden sorban egy tesztnek megfelelő válasz (DA vagy NU), a bemeneti állománynak megfelelő sorrendben.

Korlátok

- $1 \leq k \leq 100$
- $1 \leq n \leq m \leq 200\,000$
- Az S típusú karakterláncok hosszának összege nem haladja meg a 200 000-t, k teszt esetén.
- A T típusú karakterláncok hosszának összege, nem haladja meg a 200 000-t, k teszt esetén.

#	Pontszám	Korlátok
1	20	$c = 1$
2	15	$c = 2$ Az S típusú karakterlánc betűi növekvő (ábécé) sorrendben vannak rendezve, az összes teszt esetén.
3	25	$c = 2$ A T típusú karakterláncok átalakíthatók a megfelelő S típusú karakterláncra kizárólag 1-es típusú művelettel.
4	40	$c = 2$ egyéb korlátozások nélkül.

Példák

opsir.in	opsir.out	Magyarázat
1 3 2 4 cc cbbd 3 2 aab aa 2 2 ac da	3 b T c S d T 2 a T b S 3 a T c S d T	Az első teszt esetén 3 különböző betű van, a követelménynek megfelelően: a b betű többször fordul elő a T -ben, a c többször fordul elő az S -ben, és a d csak a T -ben fordul elő.
2 1 2 2 zx zx	DA	A karakterláncok azonosak anélkül, hogy bármilyen műveletet kellene alkalmazni.
2 2 2 3 ab bca 4 4 bacc cbac	DA NU	Az első teszt esetén a T karakterláncot rendezzük, így "<u>abc</u>" lesz. Ezután töröljük a harmadik betűt, így az S-el egyenlő karakterláncot kapjuk. A második teszt esetén, ha 2-es típusú műveletet alkalmazunk az első 2 betűből álló részkarakterláncra, megkapjuk a "<u>bcac</u>"-t. Tekintettel arra, hogy az első 2 betű színes lett a rendezés után (a színes betűket aláhúzással jelöljük), többé nem alkalmazhatunk 2-es típusú műveletet a "<u>ca</u>" részkarakterláncra. Ezért nem tudjuk átalakítani a T karakterláncot úgy, hogy S-el legyen egyenlő.

Feladat Poseidon

Bemenet poseidon.in
Kimenet poseidon.out

A szigetek varázslatos tartományában évente kincsvadászatot tartanak, ahol a csapatok fel kell fedezzék a vízzel körülvett elvarázslott szigeteket. A király parancsára minden szigeten kincseket rejtene el. A tartomány térképe egy $n \times m$ -es méretű mátrix formájában van ábrázolva, melynek elemei 1 méteres oldalú négyzet alakú területek kódjai, a következőképpen:

- vizet tartalmazó terület, -1 -el jelölve;
- csak földet tartalmazó terület, 0 -val jelölve; vagy
- földet és egyetlen kincset tartalmazó terület, egy nem nulla természetes számmal jelölve.

Két terület szomszédos egymással, ha közös oldala van. Két terület ugyanahhoz a szigethez tartozik, ha szomszédosak, vagy ha az egyik területről a másikra olyan úton juthatunk el, amely több területen halad keresztül, és bármelyik két egymást követő terület ezek közül szomszédos. Idén Poseidon kapitány meg akarja viccelni A. királyt, és permutálja a kincseket úgy, hogy minden kincs egy olyan területre kerüljön, ahol egy másik kincs volt kezdetben. Azonban, hogy kevésbé legyen feltűnő, a kincsek ott maradnak azon a szigeten, ahol eredetileg voltak.

Követelmények

Először is, Poseidon kapitány a következő követelményeket kívánja megoldani:

1. Hány kincs van azon a szigeten, ahol Poseidon kapitány van?
2. Hány megoldás létezik arra, hogy az összes kincset úgy helyezzük el, hogy minden kincs ugyanazon a szigeten belül kerüljön át egy olyan területre, ahol kezdetben egy másik kincs volt? Mivel a megoldások száma nagyon nagy lehet, a választ modulo $1\,000\,000\,007$ -el kell megadni.

Bemeneti adatok

A poseidon.in bemeneti állomány első sora egy c természetes számot tartalmaz, a megoldandó követelmény számát (1 vagy 2).

A második soron, két természetes szám található, az n és az m , ebben a sorrendben, a következő jelentéssel: a szigetek varázslatos tartományának térképének a sorainak a száma, illetve az oszlopainak a száma. A következő n sor mindegyikében m darab egész szám található, a mátrix területeinek kódjaival. Ha $c = 1$, akkor az állomány utolsó sorában van még két természetes szám, az x_p és y_p , annak a területnek a sorszáma és az oszlopszáma, amelyen Poseidón található.

Kimeneti adatok

A poseidon.out kimeneti állomány első sora egy természetes számot tartalmaz: ha $c = 1$, akkor az 1-es követelményben meghatározott kincsek összlétszáma, ha $c = 2$, akkor a 2-es követelményben meghatározott elhelyezési megoldások számát, a megjelölt formában.

Korlátok

- $1 \leq c \leq 2$
- $1 \leq n, m \leq 1000$
- A mátrix elemei egész számok, a $[-1, n \cdot m]$ tartományból. A mátrix 1-től van indexelve.
- Bármely két különálló terület, amely egy-egy kincset tartalmaz, különböző számokkal lesz jelölve.
- A szigetek tömör földterületekből állnak és garantáltan nem tartalmaznak vizet a belsejükben.
- $1 \leq x_p \leq n, 1 \leq y_p \leq m$, az x_p és y_p által jelölt terület tartalmaz földet.
- n_c -el jelöljük a kincsek maximális számát egyetlen szigeten.

#	Pontszám	Korlátok
1	5	$c = 1$ és egyetlen sziget létezik
2	10	$c = 1$ és $n = 1$
3	26	$c = 1$, egyéb korlátozások nélkül
4	13	$c = 2$ és $n_c \leq 4$
5	17	$c = 2$ és $n_c \leq 8$
6	29	$c = 2$, egyéb korlátozások nélkül

Példák

poseidon.in	poseidon.out	Magyarázat
1 8 6 -1 -1 -1 -1 -1 -1 -1 0 0 -1 -1 -1 -1 9 1 0 -1 -1 -1 3 4 0 -1 -1 -1 0 0 -1 0 -1 -1 -1 -1 5 10 -1 -1 -1 0 6 7 -1 -1 -1 -1 -1 -1 -1 2 2	4	Két sziget van, az egyik az 1,3,4,9 kincseket tartalmazza, és a másik az 5,6,7,10 kincseket tartalmazza. Poseidon azon a szigeten van, amely az 1,3,4,9 kincseket tartalmazza.
2 8 6 -1 -1 -1 -1 -1 -1 -1 0 0 -1 -1 -1 -1 9 1 0 -1 -1 -1 3 4 0 -1 -1 -1 0 0 -1 0 -1 -1 -1 -1 5 0 -1 -1 -1 0 6 8 -1 -1 -1 -1 -1 -1 -1	18	A kincsek átrendezésének 18 módja van. Ezek egyike: -1 -1 -1 -1 -1 -1 -1 0 0 -1 -1 -1 -1 3 9 0 -1 -1 -1 4 1 0 -1 -1 -1 0 0 -1 0 -1 -1 -1 -1 6 0 -1 -1 -1 0 8 5 -1 -1 -1 -1 -1 -1 -1