

KString



СЕЗОН 2021/2022 – ЧЕТВЪРТИ РУНД

Един низ от букви наричаме **К**-симетричен, ако може да бъде представен като **К** слепени копия на някакъв низ. Например низът "abababab" е едновременно **1**-симетричен ($1 \times$ "abababab"), **2**-симетричен ($2 \times$ "abab") и **4**-симетричен ($4 \times$ "ab"), но не е **3**-симетричен или **6**-симетричен. Очевидно всеки низ е **1**-симетричен.

Даден е низ **S**, състоящ се от малки латински букви и естествено число **K**. Вашата задача е да пренаредите буквите в низа **S** по такъв начин, че полученият низ да стане **К**-симетричен.

Вход

От единствения ред на файла `kstring.in` се въвеждат низът **S**, съставен от малки латински букви, и числото **K**.

Изход

На един ред на файла `kstring.out` изведете пренаредените букви на **S**, така че да образуват **К**-симетричен низ, или "-1", ако е невъзможно. Ако има няколко решения, изведете кое да е от тях.

Ограничения

$$1 \leq |S| \leq 10^5$$

$$1 \leq K \leq |S|$$

Ограничение по време: 0.2 сек.

Ограничение по памет: 256 МВ.

Примерни тестове

Вход (<code>kstring.in</code>)	Изход (<code>kstring.out</code>)
abacbc 2	abcabc

Вход (<code>kstring.in</code>)	Изход (<code>kstring.out</code>)
abbaba 3	bababa

Вход (<code>kstring.in</code>)	Изход (<code>kstring.out</code>)
abccaba 2	-1

Хари много обича да се разхожда във Варна. За да сме точни – в квартал „Чайка“. Докато Хари планираше следващото си пътуване, разбра, че строителна фирма „Гълъб“ е започнала строежа на нови сгради в квартала. Това въобще не му се хареса, защото по този начин ще се развали градският пейзаж. За щастие, Бари, приятел на Хари, работи във фирмата и има плановете за строежа.

Кварталът може да се представи като матрица $N \times M$. Редовете и колоните са индексирани от 1. Бари знае позициите на всяка от предвидените Q сгради. Сграда с номер s ще бъде построена в клетка (x_s, y_s) (може да бъде построена повече от една сграда в една и съща клетка). Всяка клетка има определен коефициент на красота на гледката $a_{i,j}$. Сграда в клетка (x, y) намалява красотата на всички клетки в нейния ред и колона с разстоянието до клетката. Разстоянието между две клетки на един ред (x, y_1) и (x, y_2) е $|y_1 - y_2|$, а между две клетки в една колона (x_1, y) и (x_2, y) е $|x_1 - x_2|$.

Хари иска да знае какъв ще е коефициентът на гледката за всяка една клетка, след като бъдат построени всички сгради. В момента той е зает да изготвя списък с местата, които иска да посети (за пореден път) във Варна, затова няма време за тези прости сметки. Той има нужда от вашата помощ. Напишете програма `scenery.cpp`, която да пресметне коефициента за всяка една клетка от квартала.

Вход

От първия ред на файла `scenery.in` се въвеждат три цели положителни числа – N , M и Q , съответно размерите на квартала и броя сгради, които ще бъдат построени. От следващите N реда се въвеждат по M числа – началният коефициент на гледката за всяка една клетка. На последните Q реда се въвеждат по две числа – реда и колоната за съответната сграда.

Изход

Във файла `scenery.out` трябва да отпечатате N реда с по M числа, разделени с интервал, които представляват коефициента на гледката за всяка от клетките в квартала, след като бъдат построени всички сгради.

Ограничения

$$1 \leq N, M \leq 1000$$

$$1 \leq Q \leq 10^5$$

$$-10^9 \leq a_{i,j} \leq 10^9$$

Scenery

СЕЗОН 2021/2022 – ЧЕТВЪРТИ РУНД



Ограничение по време: 2sec.

Ограничение по памет: 256 MB.

Примерен тест

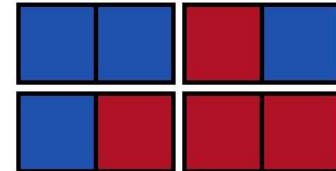
Вход (scenery.in)	Изход (scenery.out)
4 5 5	2 7 -10 12 -1
3 9 -4 15 2	8 -9 -6 8 0
11 -5 -2 8 1	4 -1 12 -13 12
7 2 14 -9 17	1 7 -9 1 -9
3 12 -8 4 -7	
2 4	
3 3	
1 2	
3 2	
4 3	

Dominoes

СЕЗОН 2021/2022 – ЧЕТВЪРТИ РУНД

Сашка не се спира да играе с любимите си доминота! Тя разполага с точно 4 вида доминота (Фиг. 1). Те имат две страни, съответно лява и дясна, като всяка от двете страни е оцветена в синьо или червено. За удобство, Сашка ги обозначила по следния начин:

- Домино №1: синьо-синьо
- Домино №2: синьо-червено
- Домино №3: червено-синьо
- Домино №4: червено-червено



Фиг. 1

Забележете, синьо-червено е различно от червено-синьо.

Сашка изважда всички доминота, които има, и ги подрежда в редица. Една *красива* редица от доминота за Сашка е такава, в която всеки две съседни страни на доминота са различни по цвят. По-точно казано, ако лявата страна на i -тото домино е l_i , а дясната страна – r_i , то за всяко $1 \leq i \leq N - 1$, трябва $r_i \neq l_{i+1}$. Примерна *красива* редица може да видите отдолу (Фиг. 2).



Фиг. 2

В България има T комплекта доминота, като i -тият комплект съдържа $d_{i,1}$ доминота от първи вид, $d_{i,2}$ от втори вид, $d_{i,3}$ от трети вид и $d_{i,4}$ от четвърти вид. Сашка първоначално приемала *междинната* оценка на един комплект доминота за броя различни красиви редици, които може да направи с тях. Две редици са различни, ако на една и съща позиция стоят различни по вид доминота. Тя осъзнала, че бройката на тези редици би могла да бъде прекалено голяма, заради това тя сменила начина на оценяване. Вместо това, тя приема, че *междинната* оценка на комплект доминота е остатъкът на броя на постижимите красиви редици от комплекта при деление с $10^9 + 7$. Колкото по-голям е този остатък, толкова по-добра е *междинната* ѝ оценка за комплекта, но това не е всичко. Сашка е склонна да преоцвети до K_i доминота от i -тия комплект, така че да повиши оценката за него. Преоцветяване на домино от i -тия комплект протича по следния начин: избира два вида доминота x и y ($x \neq y$, $1 \leq x, y \leq 4$, $d_{i,x} \neq 0$) и намалява $d_{i,x}$ с 1 и увеличава $d_{i,y}$ с 1 ($d_{i,x} := d_{i,x} - 1$, $d_{i,y} := d_{i,y} + 1$, като $:=$ е обозначено знак за присвояване). **Така, в крайна сметка *крайната* оценка на комплект е максималната междинна оценка от всички възможни преобоядисвания.** Сашка иска да намери тази крайна оценка за всеки един комплект, но трудно би я намерила ръчно, поради което, като неин трети братовчед, сте нает със задачата да напишете програма `dominoes.cpp`, която да я намери.

Dominoes



СЕЗОН 2021/2022 – ЧЕТВЪРТИ РУНД

Вход

На първия ред от файла `dominoes.in` се въвежда положителното число T – броят комплекти доминота в България. На всеки i -ти от следващите T реда се въвеждат 5 естествени числа, съответно $d_{i,1}, d_{i,2}, d_{i,3}, d_{i,4}, K_i$.

Изход

За всеки от комплектите изведете на отделен ред във файла `dominoes.out` крайната му оценка.

Ограничения

$$1 \leq T \leq 5$$

$$1 \leq d_{i,1} + d_{i,2} + d_{i,3} + d_{i,4} \leq 120$$

$$0 \leq d_{i,1}, d_{i,2}, d_{i,3}, d_{i,4} \leq 120$$

$$0 \leq K_i \leq 20$$

$$\text{За всеки } 1 \leq i, j \leq T, d_{i,1} + d_{i,2} + d_{i,3} + d_{i,4} = d_{j,1} + d_{j,2} + d_{j,3} + d_{j,4}$$

Ограничение по време: 1 сек.

Ограничение по памет: 256 МВ.

Примерни тестове

Вход (<code>dominoes.in</code>)	Изход (<code>dominoes.out</code>)
3 1 1 1 0 0 1 0 1 1 0 3 0 0 0 2	1 3 3
3 3 3 3 1 3 8 1 1 0 4 10 0 0 0 5	120 40 5
2 9 2 5 10 5 9 8 4 5 3	3171168 2944656
3 30 30 30 30 20 30 60 15 15 20 60 60 0 0 20	999079969 997861536 0

Dominoes

СЕЗОН 2021/2022 – ЧЕТВЪРТИ РУНД



Обяснение на примерните тест

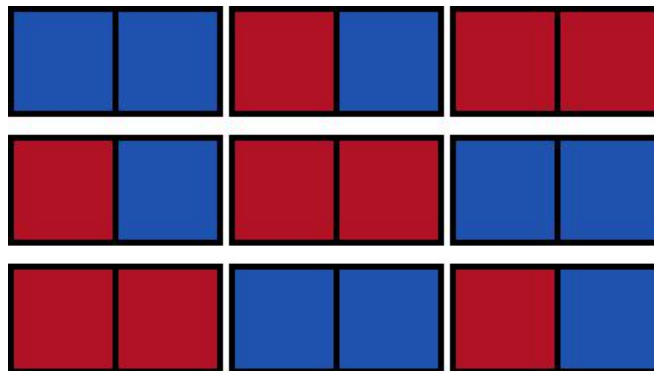
За първия примерен тест:

- За първия комплект доминота единствената възможна редица е:



Фиг. 3

- За втория комплект доминота трите възможни редици са следните:



Фиг. 4

- За третия пример може да се стигне от $\{3,0,0,0\} \rightarrow \{1,0,1,1\}$

Останалите примерни тестове имат вероятно много красиви редици, но мястото тук не би стигнало за тях 😊.