

**Нов български университет**  
**Департамент Информатика**  
Школа „Състезателно програмиране“  
СЪСТЕЗАНИЕ, 24 март 2012 г.

**Задача А. Охрана**

Охранителна фирма трябва да охранява  $n$  обекта ( $n < 15$ ), номерирани с числата от 1 до  $n$ . За всеки ден се изготвя график, в който е записано кой от служителите на фирмата кой номер обект охранява. Графикът е низ, в който всеки служител е отбелязан с по една латинска буква – първата буква от името му, като охранителят назначен на първия обект е на първо място, охранителят назначен на втория обект е на второ място и т.н. Не се прави разлика между малки и големи букви. Някои от охранителите са означени с една и съща буква. След оповестяване на графика тези от служителите, които са отбелязани с една и съща буква, се договарят кой от тях, кой обект да отиде да охранява. Графикът за всеки следващ ден се получава от графика на предишния, така че е следващият лексикографски спрямо предишния, получен от същите букви. Ако не е възможно да се направи следващ лексикографски, тогава се преминава към лексикографски най-малкия график.

На служител със зададен инициал възниква проблем за вземане на дежурство след  $d$  дни. Помогнете на потенциалния му заместник, като отпечатате графика за съответния ден и намерите в него номерата на обекти, които заместникът би могъл да охранява.

От **стандартния вход** трябва да прочетете множество тестови примери. За поредния тест последователно ще бъдат зададени графикът с който се започва  $S$ , инициал  $x$  на служител, на който се налага да отсъства и цяло положително числото  $d$ .  $S$ ,  $x$  и  $d$  са зададени на отделни редове.

За всеки тест се извежда на един ред графика за след  $d$  дни (всичките му буквите трябва да са малки), а на следващ ред – номерата на обекти, които ще се охраняват от служител с инициала  $x$ . Следващият лексикографски низ за график, състоящ се от еднакви букви, е същият низ.

**Пример:**

Вход	Изход
acbc c 1	accb 2 3

**Нов български университет**  
**Департамент Информатика**  
Школа „Състезателно програмиране“  
СЪСТЕЗАНИЕ, 24 март 2012 г.

**Задача В. Точкомания**

Няколко приятели играят игра, която се нарича „Точкомания“. Във всеки рунд на играта, всеки играч избира една точка в равнината с целочислени координати. Резултатът на всеки един от играчите е броя на точките на противниците, които са с по-малка или равна абсциса и с по-малка или равна ордината. Играчите нямат право да избират точки, които вече са избрани от друг играч. Напишете програма, която, по зададени координати на избраните точки, намира резултата на всеки от играчите.

На първия ред на стандартния вход е зададено цяло положително число  $T$  – броят на тестовите примери. Всеки тестов пример започва с броя  $N$  на играчите ( $1 \leq N \leq 50\,000$ ). На  $i$ -тия от следващите  $N$  реда са зададени координатите  $X_i$  и  $Y_i$  на  $i$ -тия играч ( $1 \leq X_i, Y_i \leq 100\,000$ ).

За всеки тестов пример, на отделен ред на стандартния изход, програмата трябва да изведе резултата на един от играчите, като резултата на  $i$ -тия играч е на  $i$ -тия ред.

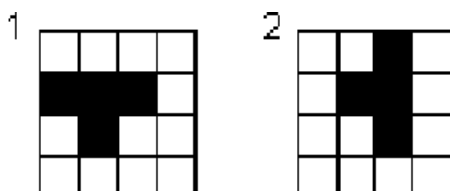
**Пример:**

Вход	Изход
2	0
3	1
1 1	2
2 2	2
3 3	1
5	3
7 1	0
5 1	1
5 5	
1 1	
3 3	

**Нов български университет**  
**Департамент Информатика**  
 Школа „Състезателно програмиране“  
 СЪСТЕЗАНИЕ, 24 март 2012 г.

**Задача С. Ключ и ключалка**

Известни са формата на ключ и ключалка, зададени като квадратни двумерни таблици от 0 и 1 (отворът на ключалката и сечението на ключа се задават с 1). Да се напише програма, която проверява дали зададеният ключ съответства на ключалката, като ключът може да е завъртян съответно на 0°, 90°, 180° и 270° (ключът може да се върти, но не може да се мести по хоризонтал и вертикал). Например, ключът от Рис. 1 може да съвпадне с ключалката от Рис. 2, ако бъде завъртян на 90°.



Програмата трябва да обработи няколко тестови примера. За всеки от тях се въвежда цяло положително число  $n$  ( $n \leq 10$ ) – размер на таблиците за ключа и ключалката. Следват  $n$  реда с по  $n$  числа (0 или 1) за ключа и  $n$  реда от по  $n$  числа (0 или 1) за ключалката. Ако  $n$  е по-малко или равно на 0, то това е края на входа.

За всеки тестов пример програмата трябва да изведе на отделен ред на стандартния изход резултата – YES, ако ключът съвпада с ключалката и NO, ако ключът не съвпада с ключалката.

**Пример:**

Вход	Изход
4	YES
0000	NO
1110	
0100	
0000	
0010	
0110	
0010	
0000	
3	
000	
010	
011	
000	
100	
110	
-1024	

**Нов български университет**  
**Департамент Информатика**  
Школа „Състезателно програмиране“  
СЪСТЕЗАНИЕ, 24 март 2012 г.

**Задача D. Бирено парти**

Смарти е програмист, който предпочита да си лежи или да пие бира, което е второто най-важно нещо в живота му (и чак тогава идва ред на програмирането). Един ден той бил поставен пред следния проблем. На бирено парти домакините били подредили в много дълга редица  $N$  бутилки бира от  $K$  различни вида. Тъй като бил колекционер на празни бирени бутилки и нямал в колекцията си нито една бутилка от тези  $K$  вида, той трябвало да направи нещо, за да си ги занесе в къщи. Но единственият начин да си вземе някакво количество бирени бутилки бил, да избере една непрекъсната поредица от тях и да ги изпие. Тъй като е много мързелив и му е много трудно да отваря бутилките, решил да избере най-късата непрекъсната поредица от бутилки, която съдържа поне по една от всичките  $K$  различни вида. Смарти е вече достатъчно пьян (нали е на бирено парти) и не може да си спомни алгоритъма, който решава тази задача. Затова ще трябва да му помогнете, като напишете програма, която определя позициите в редицата на първата и последната бутилки от най-късата непрекъсната поредица, изпълняваща условието. Ако има няколко такива поредици, програмата трябва да намери тази, която се среща най-рано в редицата.

Програмата трябва да прочете от първия ред на стандартния вход броят  $T$  на ситуацияите, които трябва да обработи. Данните за всяка ситуация са в два реда на стандартния вход. На първия са записани числата  $N$  и  $K$  ( $1 \leq N \leq 10000000$ ,  $1 \leq K \leq 13000$ ,  $N.K \leq 170000000$ ) разделени с интервал, а на втория –  $N$  числа (от 1 до  $K$ ), разделени с по един интервал.

За всяка ситуация програмата трябва да изведе на стандартния изход две числа – номерата на началната и крайната бутилки в поредицата.

**Пример:**

Вход	Изход
2	3 5
5 3	2 5
1 3 3 2 1	
5 3	
1 1 2 2 3	

**Нов български университет**  
**Департамент Информатика**  
Школа „Състезателно програмиране“  
**СЪСТЕЗАНИЕ, 24 март 2012 г.**

**Задача Е. Необикновена сума**

Нека са дадени две обикновени дроби  $a/b$  и  $c/d$ . Като ги съберем, получаваме обикновена дроб, която преобразуваме до несъкратимата  $e/f$ . След това намираме сумата от делителите на числителя  $e$  и делителите на знаменателя  $f$  и означаваме тази сума с  $S$ . Така, за всяка двойка обикновени дроби, можем да намерим съответната сума  $S$ . Например, нека дробите са  $2/5$  и  $3/6$ . Сборът им е  $9/10$ , делителите на 9 са 1, 3 и 9, а на 10 са 1, 2, 5 и 10. Затова сумата  $S = 1 + 3 + 9 + 1 + 2 + 5 + 10 = 31$ .

Напишете програма, която по зададени две обикновени дроби намира и извежда търсената сума  $S$ .

Програмата трябва да може да обработи няколко тестови примера. На първия ред на **стандартния вход** ще бъде зададен броят на тестовите примери. За всеки тест, от един ред се въвеждат четири цели положителни числа, разделени с интервал – съответно числителят  $a$  и знаменателят  $b$  на първата дроб, и числителят  $c$  и знаменателят  $d$  на втората. Всяко от числата не надвишава 10 000.

За всеки тест, на **стандартния изход** програмата трябва да изведе едно цяло число – търсената сума  $S$ .

**Пример:**

<b>Вход</b>	<b>Изход</b>
2	5
1 4 1 12	31
2 5 3 6	

**Нов български университет**  
**Департамент Информатика**  
**Школа „Състезателно програмиране“**  
**СЪСТЕЗАНИЕ, 24 март 2012 г.**

**Задача F. Бирарии**

Смарти посещава  $N$  бирарии, номерирани от 1 до  $N$ , чието местоположение се определя от координатите им в правоъгълна координатна система. Бирариите са разделени в две непразни множества  $A$  и  $B$  и са свързани с  $M$  двупосочни улици, всяка от които свързва две различни бирарии. За всяка улица е известно времето за преминаване от единия до другия ѝ край. За всяка бирария от множеството  $A$  съществува път от нея до бирария от множеството  $B$ . Всяка вечер Смарти е на бирено парти в някоя от бирариите от множеството  $A$ . След като пийне порядъчно, той иска да си допие в някоя бирария от множеството  $B$ . Напишете програма, която за всяка бирария от множеството  $A$  да определи минималното време, необходимо за предвижването от нея до някоя бирария от множеството  $B$ . Смарти ще ви е много благодарен.

На първия ред на **стандартния вход** ще бъде зададен броят на тестовете. Всеки тест започва с ред, съдържащ числата  $N$  и  $M$  ( $N \leq 10^3$ ). Следват  $N$  реда с координатите на бирариите, подредени в нарастващ ред на номерата им, както и “ $A$ ” или “ $B$ ”, определящо принадлежността на бирария към съответното множество. Следват  $M$  реда с описания на улиците – двата края на улицата и времето за преминаването през нея. Всички координати са цели числа в интервала  $[-10^4, 10^4]$ , а времената – положителни цели, не надхвърлящи  $10^4$ .

За всеки пример трябва да изведете на **стандартния изход** толкова реда, колкото са бирариите от множеството  $A$ . На всеки от тези редове трябва да има по две числа – номер на бирария от множеството  $A$  и минималното време, за предвижването от нея до някоя бирария от множеството  $B$ . Редовете трябва да са подредени по нарастващ ред на номерата на бирариите.

**Пример:**

Вход	Изход
1	1 10
8 8	3 21
-1 4 A	5 18
2 0 B	8 3
4 -1 A	
2 3 B	
4 4 A	
0 0 B	
0 3 B	
-1 -1 A	
1 6 10	
2 3 21	
2 4 2	
2 6 20	
4 7 7	
4 5 18	
6 8 3	
6 7 6	

***Забележка:** За тези, които не решат задачата, журито приемаме, че не искат да помагат на пияници!*

**Нов български университет**  
**Департамент Информатика**  
Школа „Състезателно програмиране“  
СЪСТЕЗАНИЕ, 24 март 2012 г.

**Задача G. Аритметични изрази**

Да се намери с колко различни аритметични изрази, съдържащи само цели числа и знак за събиране е възможно едно дадено цяло положително число  $N$  да се представи като сума на цели положителни числа, всяко от които е по-малко или равно на  $K$

**Ограничения:**  $0 < K < N < 40$ .

**Вход:** Брой на тестовите примери, след който за всеки тест на отделен ред са дадени стойностите на  $N$  и  $K$ , разделени с един интервал.

**Изход:** За всеки тест да се изведе на отделен ред съответният търсен брой.

**Пример:**

Вход	Изход
3	3
3 2	1
4 1	7
4 3	

**Пояснение:**

$$3 = 1+1+1 = 2+1 = 1+2;$$

$$4 = 1+1+1+1;$$

$$4 = 1+1+1+1=1+1+2=1+3=3+1=1+2+1=2+1+1=2+2.$$

**Нов български университет**  
**Департамент Информатика**  
Школа „Състезателно програмиране“  
СЪСТЕЗАНИЕ, 24 март 2012 г.

**Задача Н. Квадрати**

Разглеждаме квадратите в равнината с лице по-малко или равно на  $S$  и с целочислени координати на върховете им. Напишете програма, която намира колко са различните квадрати от описания вид (считаме, че два квадрата са различни тогава и само тогава, когато имат различни лица).

**Ограничения:**  $0 < S < 20\,000\,000$ .

**Вход:** Брой на тестовите примери, след който на отделни редове за всеки тестов пример е записана стойността на  $S$ .

**Изход:** На отделен ред за всеки тестов пример да се изведе съответният търсен брой.

**Пример:**

Вход	Изход
2	2
2	13
25	

**Пояснение за първи тест:** два различни квадрата са например тези с координати на върховете  $(0,0)$ ,  $(0,1)$ ,  $(1,1)$ ,  $(1,0)$  и  $(0,0)$ ,  $(1,1)$ ,  $(0,2)$ ,  $(-1,1)$ . Всички други квадрати, които имат лице по-малко или равно на 2 и са с целочислените координати на върховете си, са еднакви с някой от описаните два квадрата.



**Нов български университет**  
**Департамент Информатика**  
Школа „Състезателно програмиране“  
СЪСТЕЗАНИЕ, 24 март 2012 г.

**Задача I. Младия земеделец**

Бай Колю, кризата го ударила и решил да става земеделец. Насадил си той домати в градината, ама взели да идват разни пришълци и да му ги крадат. Решил да си купи мрежа да си направи ограда, ама броил, броил стотинките и стигнал до извода, че трябва да си оптимизира разхода, като загради възможно най-оптимално своите домати. За колове на оградата щял да ползва колците на доматиите.

Помогнете му като напишете програма, която намира колко метра телена ограда трябва да купи (в магазина продават само по цял метър), за да загради всичките си домати.

**Входът** съдържа много тестови примери, като за всеки от тях се посочва броя домати N в градината му и последователно се дават техните реални координати x и y. За край на тестовете се въвежда N = 0.

На стандартния **изход** се извежда по едно цяло число на ред за всеки тестов пример, представляващо намерената дължина на мрежа, която Колю трябва да си купи.

**Пример:**

Вход	Изход
6 1 1 4 1 4 5 1 5 2 3 3 2 0	14

**Нов български университет**  
**Департамент Информатика**  
Школа „Състезателно програмиране“  
СЪСТЕЗАНИЕ, 24 март 2012 г.

**Задача J. Transformers**

Нещото се представяло като фигура от  $N$  ( $N \leq 1\,000\,000\,000$ ) последователно вързани точки в двумерното пространство. Това нещо се превръщало в друго нещо за да заблуди другите неща, чрез редица от трансформации ( $M$  на брой):

1. Транслация с вектор  $V$  ( $V_x, V_y$ ). [1  $V_x V_y$ ]
2. Ротация на ъгъл  $\alpha$  (в градуси) спрямо точка  $P$  ( $P_x, P_y$ ) [2  $\alpha P_x P_y$ ]
3. Симетрия спрямо права  $Y = kX + n$  [3  $k n$ ]
4. Мащабиране по осите  $X$  и  $Y$  с коефициенти  $K_x$  и  $K_y$  [4  $K_x K_y$ ]

**Входът** съдържа много тестови примери, като за всеки от тях се посочва броя точки  $N$  на нещото. На всеки следващ ред от теста се подават техните реални координати  $x$  и  $y$ . Следващият ред задава броя трансформации  $M$  и всяка от тях се описват номера на трансформацията и съответните параметри. За край на тестовете се въвежда  $N = 0$ .

На стандартния **изход** за всеки тестов пример да се изведат новите координати на точките на нещото.

**Пример:**

Вход	Изход
4	2 2
1 1	4 2
1 3	2 5
4 1	4 5
4 3	
2	
1 1 1	
3 1 0	
0	