



**Департамент Информатика**  
**Школа „Състезателно програмиране“**  
**СЪСТЕЗАНИЕ, 24 ноември 2012 г.**

**Задача А. Туристи**

$N$  туристи, номерирани от 1 до  $N$  ( $1 \leq N \leq 50\,000$ ), трябва да бъдат разпределени в хотели. Желание да бъдат в един хотел са заявили  $M$  ( $1 \leq M \leq 150\,000$ ) двойки туристи, и такива туристи трябва да бъдат поставени в един и същ хотел. Ако турист не е заявил желание да бъде в хотел заедно с друг турист, той трябва да бъде сам в хотела си. Напишете програма, която да намира броя на хотелите.

**Вход**

На първия ред на стандартния вход е зададен броят на тестовете. За всеки тест, на първия ред ще бъдат зададени, разделени с по един интервал, числата  $N$  и  $M$ . На всеки от следващите  $M$  реда – номерата на двама туристи, желаещи да бъдат в един обор.

**Изход**

За всеки тест програмата трябва да изведе на отделен ред на стандартния изход търсения брой на хотелите.

**Пример:**

Вход	Изход
1 4 2 1 3 4 3	2



**Департамент Информатика**  
**Школа „Състезателно програмиране“**  
**СЪСТЕЗАНИЕ, 24 ноември 2012 г.**

**Задача В. Минимум и максимум**

Разглеждаме квадрат с размери  $N$  на  $N$ , разделен на квадратчета със страна 1. За всяко от квадратчетата е определена цена  $M_{ij}$  (цяло число). Целта е, първо, да се премине от най-долното ляво квадратче до най-горното дясно, като на всяка стъпка може да се преминава от текущото квадратче, в квадратчето отдясно, в квадратчето отгоре или в квадратчето вдясно отгоре по диагонал. За всяко преминато квадратче, цената  $M_{ij}$  се добавя като печалба. Целта е, разбира се, печалбата да е максимална. След това, трябва да се върнем обратно в началното квадратче. При това може да се придвижваме от текущото квадратче само в квадратчето отляво, в квадратчето отдолу или в квадратчето вляво отдолу по диагонал. За всяко преминато при връщането квадратче, трябва да се плати цената му  $M_{ij}$ . Целта е, сумата от цените на квадратчетата, преминати при връщането да е минимална.

Напишете програма, която да намира максималната печалба на отиване и минималната загуба на връщане.

Програмата трябва да може да обработи няколко тестови примера при едно извикване. На първия ред на **стандартния вход** ще бъде зададен броят на тестовите примери. За всеки тест, на първия ред ще бъде зададено числото  $N$  ( $3 \leq N \leq 500$ ). Следват  $N$  реда с по  $N$  цели числа  $M_{ij}$  – цените на съответните квадратчета ( $-100 \leq M_{ij} \leq 100$ ). Цената на най-горното най-дясно квадратче (последното число в първия ред) и на най-долното най-ляво квадратче (първото число на последния ред) винаги е нула.

За всеки тест, на един ред на **стандартния изход** програмата трябва да изведе намерените максимална цена на отиването и минимална на връщането, разделени с един интервал.

Вход	Изход
1 4 6 1 2 0 4 4 1 5 1 6 3 2 0 0 2 7	17 4



# НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

София 1635, ул. Монтевидео 21  
 тел.: 55 81 37, 55 21 35, факс: 957 19 30

## Департамент Информатика

### Школа „Състезателно програмиране“

СЪСТЕЗАНИЕ, 24 ноември 2012 г.

### Задача С. Матрици

Разполагаме със следните 5 вида 4x4 матрици:

.....	****	****	****	****
.....	.....	****	****	****
.....	.....	.....	****	****
.....	.....	.....	.....	****

Както и с една по-голяма матрица, състояща се от  $M \times N$  от посочените по-горе матрици. За всеки от посочените 5 вида матрици се интересуваме колко пъти се среща в по-голямата матрица.

На първия ред на стандартния вход е зададен броят на тестовете. За всеки тест, на първия ред ще бъдат зададени, разделени с по един интервал, числата  $M$  и  $N$  ( $1 \leq M, N \leq 100$ ). Следват  $5M + 1$  реда, всеки съдържащ  $5N + 1$  символа. Всяка подматрица е заградена изцяло със символа '#’.

За всеки тест програмата трябва да изведе на отделен ред на стандартния изход 5 числа, като поредното показва броя на срещанията на съответната матрица.

Вход	Изход
<pre> 2 1 2 ##### #....#***# #....#***# #....#....# #....#....# ##### 2 3 ##### #***#***#***# #***#***#***# #***#....#***# #....#....#***# ##### #....#***#***# #....#***#....# #....#....#....# #....#....#....# #####                     </pre>	<pre> 1 0 1 0 0 1 1 2 1 1                     </pre>



**Департамент Информатика**  
**Школа „Състезателно програмиране”**  
**СЪСТЕЗАНИЕ, 24 ноември 2012 г.**

**Задача D. Игра**

Както може би ви е известно, в популярната игра “Blackjack” целта е да събереш карти, чиято сума е максимално близка до 21. В играта, която ние ще играем, играча е вашата програма и ще трябва на всяка врътка да изчита по  $N$  цели числа ( $3 \leq N \leq 100$ ). От тези числа, ще трябва да избере три, така че, сумата на тези числа да е максимално близка, но не надхвърляща зададено число  $M$  ( $10 \leq M \leq 300\,000$ ).

Всяка от врътките е зададена на стандартния вход и се състои от два реда. На първия ред ще бъдат зададени  $N$  и  $M$ . Втория ред ще съдържа  $N$ -те числа. Край на входа е число, не отговарящо на ограниченията за  $N$ .

За всяка от врътките програмата трябва да изведе на отделен ред на стандартния изход търсената максимална сума, отговаряща на изискванията.

**Пример:**

Вход	Изход
5 21	21
5 6 7 8 9	497
10 500	
93 181 245 214 315 36 185 138	
216 295	
0	


**Департамент Информатика**
**Школа „Състезателно програмиране“**
**СЪСТЕЗАНИЕ, 24 ноември 2012 г.**
**Задача Е. Математически израз**

По зададен валиден математически израз със скоби, трябва да намерите всички различни валидни изрази, които могат да бъдат получени, след премахването на двойка (отваряща - затваряща) скоби от оригиналния израз. Два израза са различни, ако имат символ по който се различават. Нека е даден израза  $(2+(2*2)+2)$ . От него можем да получим следните изрази  $(2+2*2+2)$ ,  $2+(2*2)+2$  и  $2+2*2+2$ .  $(2+2*2)+2$  и  $2+(2*2)+2$  не са валиден отговор, тъй като те биха били получени, след премахването на двойки скоби, които не са валидни. Повече от една двойка скоби могат да ограждат една и съща част от израза!

На първия ред на стандартния вход е зададен броят на тестовете. За всеки тест, на отделен ред е зададен самия израз. Този израз ще е коректен математически израз, състоящ се от цели положителни числа. Аритметичните операции са представени чрез символите  $+$ ,  $*$ ,  $-$  и  $/$ , а скобите с  $($  и  $)$ . Дължината на израза няма да надвишава 200 символа и ще съдържа поне една и не повече от 10 двойки скоби.

За всеки пореден тест програмата трябва да изведе на отделен ред на стандартния изход низа „test case # x:“, следван от празен ред, където x е номера на поредния тест. Следват всички валидни изрази, които могат да бъдат получени, сортирани лексикографски. Преди изхода за всеки тест, без първия трябва да има точно по един празен ред. Края на целия изход не трябва да съдържа празен ред.

Вход	Изход
3 (0/(0)) (2+(2*2)+2) (1+(2*(3+4)))	test case # 1:  (0/0) 0/(0) 0/0  test case # 2:  (2+2*2+2) 2+(2*2)+2 2+2*2+2  test case # 3:  (1+(2*3+4)) (1+2*(3+4)) (1+2*3+4) 1+(2*(3+4)) 1+(2*3+4) 1+2*(3+4) 1+2*3+4



**Департамент Информатика**  
**Школа „Състезателно програмиране”**  
**СЪСТЕЗАНИЕ, 24 ноември 2012 г.**

**Задача F. Подредица**

Дадена е редица от  $n$  различни положителни цели числа, по-малки от  $10^{12}$ . Да се намери подредица, образувана от най-големите  $m$  числа в редицата ( $m < n$ ). Ако един елемент на дадената редица е преди друг елемент (има по-малък индекс), то той е преди него и в получената подредица.

**Вход**

За един пример от стандартния вход се въвеждат две числа –  $n < 10^{12}$  и  $m < 1000$  и след това  $n$  елемента на дадената редица. Входът съдържа много примери.

**Изход**

За всеки пример от входа се отпечатва получената подредица на един ред на стандартния изход.

**Пример:**

<b>Вход</b>	<b>Изход</b>
4 2	23 5
1 23 4 5	8 7 12
6 3	
2 8 7 5 12 1	



**Департамент Информатика**  
**Школа „Състезателно програмиране”**  
**СЪСТЕЗАНИЕ, 24 ноември 2012 г.**

**Задача G. Големия Джо**

След като приключил с разбойничеството, защото все по-често шерифите започнали да хващат и бесят разбойниците, Големия Джо решил да се захване със земеделие. Отишъл в незаетите още равнини на средния запад, и като порядъчен нов заселник се представил на местния шериф:

- Искам земя, ще правя ранчо.

Шерифът обяснил накратко правилата за даване на земя:

- Отиваш в склада, получаваш колци и оградна мрежа. Колкото земя заградиш – твоя е!

Така и направил нашия приятел – взел колците и мрежата, отнесъл ги в равнината, на празно място (други огради се не се виждали наоколо) и зачудил – как да си загради най-голям участък? Помогнете на Джо!

**Вход**

На стандартния вход се задават много примери, състоящи се от по две цели числа на един ред – броя на колците (по-малък от 100) и дължината на мрежата в метри (по-малка от 1 км).

**Изход**

За всеки пример на стандартния изход да се отпечата заградената максимална площ, закръглена с точност до 1 кв.м.

**Пример:**

<b>Вход</b>	<b>Изход</b>
5 50	172
10 100	769



**НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ**

София 1635, ул. Монтевидео 21  
тел.: 55 81 37, 55 21 35, факс: 957 19 30

**Департамент Информатика**  
**Школа „Състезателно програмиране“**  
**СЪСТЕЗАНИЕ, 24 ноември 2012 г.**

**Задача N. Сума**

Дадено е цяло положително число  $N$ , записано с не повече от 10 цифри. Образуваме всевъзможните низове, получени чрез изтриване на няколко (нула, една или повече) цифри от  $N$ . Така получените низове разглеждаме като числа и означаваме сумата на всичките с  $S$ . Напишете програма, която въвежда  $N$  и извежда  $S$ .

**Вход**

От всеки ред на стандартния вход се въвежда по едно цяло число  $N$ .

**Изход**

На всеки ред на стандартния изход се извежда по едно цяло число  $S$ .

**Пример:**

<b>Вход</b>	<b>Изход</b>
123	177
101	125





Департамент Информатика  
Школа „Състезателно програмиране“  
СЪСТЕЗАНИЕ, 24 ноември 2012 г.

### Задача I. Лешояд

Ловната територия на лешояд е правоъгълна област с размери  $N \times M$  километра. Всеки квадрат (поле) от областта има надморска височина  $H_{ij}$  метра, където  $1 \leq i \leq N$  и  $1 \leq j \leq M$ . Във всеки квадрат въздухът се издига или спуска с постоянна скорост  $V_{ij}$  метра за единица време, достатъчно на лешояда да прелети през квадрата. Когато въздухът се издига, скоростта е положително цяло число, ако се спуска е отрицателна и 0, ако е в покой. Лешоядът планира, като при преминаване от едно поле в друго съседно губи височина  $k$  и печели толкова, колкото е скоростта на възходящия поток в полето от което тръгва. Съседни на полето  $(i, j)$  са полетата:

$$(i + 1, j), (i - 1, j), (i, j + 1) \text{ и } (i, j - 1).$$

Поради вродено заболяване лешоядът има страх от височина и не може да се издига на повече от  $H_{max}$  метра спрямо нивото на полето, над което планира в момента. Гнездото на лешояда се намира в поле  $(i_1, j_1)$ , а в поле  $(i_2, j_2)$  има труп на коза, загинала при неизвестни обстоятелства.

Напишете програма, която да определи най-малкият брой прелитания между полетата от областта, необходими на лешояда, за да достигне от гнездото си до трупа на козата. При всички прелитания лешоядът трябва да е строго над полето, освен в гнездото и при достигане на плячката, когато е точно на височината на полето.

### Вход

На първия ред за всеки пример са записани целите положителни числа  $M$ ,  $N$ ,  $k$  и  $H_{max}$ . Следват  $M \times N$  реда, всеки съдържащ две цели числа  $H_{ij}$  и  $V_{ij}$  в следния ред:

$$H_{1,1} \quad V_{1,1}$$

$$H_{2,1} \quad V_{2,1}$$

$$H_{3,1} \quad V_{3,1}$$

.....

$$H_{N,1} \quad V_{N,1}$$

$$H_{1,2} \quad V_{1,2}$$

$$H_{2,2} \quad V_{2,2}$$

$$H_{3,2} \quad V_{3,2}$$

.....

$H_{N,2} V_{N,2}$

.....

$H_{1,M} V_{1,M}$

$H_{2,M} V_{2,M}$

$H_{3,M} V_{3,M}$

.....

$H_{N,M} V_{N,M}$

Следва ред с координатите на гнездото  $i_1$  и  $j_1$ , а на последния ред са координатите на плячката  $i_2$  и  $j_2$ .

### Изход

На отделен ред на стандартния изход, за всеки пример програмата трябва да изведе едно цяло число, равно на търсения минимален брой прелитания за съответния пример от входа. Ако не е възможно да бъде достигната плячката, програмата трябва да изведе думата `crash`.

### Ограничения

$0 \leq M \leq 250, 0 \leq N \leq 250, 0 \leq H_{max} \leq 20, k \leq 5, 0 \leq H_{i,j} \leq 30, -5 \leq V_{i,j} \leq 5$

### Пример:

Вход	Изход
3 5 2 10 2 2 3 3 4 2 5 2 4 2 2 2 4 -2 5 -2 7 -2 5 -2 2 1 2 1 3 1 3 1 2 1 4 1 4 3	8



**НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ**

София 1635, ул. Монтевидео 21  
тел.: 55 81 37, 55 21 35, факс: 957 19 30

**Департамент Информатика**  
**Школа „Състезателно програмиране“**  
**СЪСТЕЗАНИЕ, 24 ноември 2012 г.**

**Задача J. Обобщен триъгълник на Паскал**

Обобщен триъгълник на Паскал получаваме по същия начин, както и добре известния триъгълник на Паскал, с тази разлика, че първият ред на триъгълника се състои от две произволни цели числа. Когато числата са 1, 1, получаваме истинския триъгълник на Паскал. Да се напише програма за генериране на обобщен триъгълник на Паскал.

**Вход**

На първия ред на стандартния вход за всеки пример са зададени числата  $a$  ( $0 \leq a < 10$ ) и  $b$  ( $0 \leq b < 10$ ), които образуват първия ред от един обобщен триъгълник на Паскал. На следващия ред е дадена редица от положителни числа – номер на ред в получения обобщен триъгълник на Паскал. Числата в редицата не са повече от 100 и са по-малки от 65. Входът съдържа много примери.

**Изход**

За всеки пример на един ред на стандартния изход програмата трябва да изведе максималните елементи от съответния (от входа) ред на обобщения триъгълник на Паскал.

Вход	Изход
1 1	252 2
10 2	1386 3 30
1 2	
12 2 6	

Ето първите 6 реда от обобщен триъгълник на Паскал с  $a = 1$ ,  $b = 2$ .

```
1 2
1 3 2
1 4 5 2
1 5 9 7 2
1 6 14 16 9 2
1 7 20 30 25 11 2
```



**Департамент Информатика**  
**Школа „Състезателно програмиране“**  
**СЪСТЕЗАНИЕ, 24 ноември 2012 г.**

### Задача К. Чинии

Домакиня е поканила гости и ги е настанила около кръгла маса с радиус  $R$ . Центърът на масата се намира в т.  $(0,0)$  на координатната система. На масата са разположени  $N$  кръгли чинии, всяка с център  $(x_i, y_i)$  и радиус  $r_i$  ( $1 \leq i \leq N$ ). В разгара на вечерта домакинята донася още една чиния с радиус  $q$ . Домакинята трябва да постави новата чиния така, че центърът ѝ да е върху масата и тя да не се застъпва с никоя от останалите чинии. Освен това, тя не трябва да размества другите чинии.

Напишете програма, която да изчислява дали домакинята може да постави новата чиния на масата.

#### Вход

На първия ред е даден броят на тестовете. Данните за всеки тест започват с ред, на който е записано цялото положително число  $N$ . На следващия ред са реалните числа  $R$  и  $q$ . Следват  $N$  реда, всеки съдържащ тройка реални числа  $x_i, y_i, r_i$ .

#### Изход

За всеки тест се изписва ред с думата **yes**, ако има намерена подходяща позиция за новата чиния. Ако няма къде да се постави чинията, програмата трябва да изпише **no**.

#### Ограничения

$N \leq 100, R \leq 50, q \leq 20, r_i \leq 20$

#### Пример

Вход	Изход
2	yes
4	no
5 2	
2 4 3	
2 -4 3	
-4 2 1	
-4 -2 1	
7	
5.000000 1.000000	
-1.362337 -0.375150 2.000000	
3.958683 0.928493 2.000000	
3.065528 -3.093581 2.000000	
0.428379 4.027148 2.000000	
-1.436027 -4.484373 2.000000	
-4.009918 2.791625 2.000000	
-4.453992 -2.178910 1.500000	



**Департамент Информатика**  
**Школа „Състезателно програмиране“**  
**СЪСТЕЗАНИЕ, 24 ноември 2012 г.**

**Задача L. Най-широка лодка**

В страната Водландия има  $N$  езера (номерирани от 1 до  $N$ ) и  $M$  канала между тях, като за всеки канал  $(i, j)$  е известна ширината му в метри  $W_{ij}$ . Каналите са двупосочни, ако канал свързва езера  $i$  и  $j$ , можем да се движим по него както от езеро  $i$  към  $j$ , така и от езеро  $j$  към  $i$ . Напишете програма, която да изчисли максималната ширина на лодка, която може да направи пътешествие между всеки две езера (лодка може да мине от едно езеро в друго, ако ширината ѝ е по-малка или равна на ширината на канала, свързващ езерата).

**Вход**

На първия ред е даден броят на тестовете. Всеки тест започва с ред, на който са записани целите положителни числа  $N$  и  $M$ . Следват  $M$  реда, на всеки от които са изписани три числа  $i, j$  и  $W_{ij}$ , указващи, че между езера  $i$  и  $j$  има канал с ширина  $W_{ij}$ .

**Изход**

За всеки тест програмата трябва да изведе по едно цяло число, равно на търсената максимална ширина на лодка.

**Ограничения:**  $N \leq 1000$ ,  $M \leq 100000$ ,  $W_{ij} \leq 200$

**Пример**

Вход	Изход
2	3
5 6	4
1 2 2	
1 4 3	
2 3 3	
2 5 2	
3 4 4	
4 5 5	
7 10	
4 6 3	
2 3 1	
3 7 5	
1 5 4	
1 4 7	
2 6 6	
5 7 5	
3 6 4	
2 7 7	
4 5 2	