

**XXIV НАЦИОНАЛНА ОЛИМПИАДА ПО ИНФОРМАТИКА**  
*Национален кръг, 18-20.04.2008 г.*

**Задача А1. ДИАМЕТЪР НА ГРАФ**

Даден е неориентиран граф  $G$  с  $n$  върха и  $n-1$  ребра. Графът е свързан и дължините на ребрата са цели неотрицателни числа. Означаваме с  $d(x,y)$  дължината на най-късия път между два върха  $x$  и  $y$  от  $G$ . Диаметър на графа  $G$  се нарича най-голямото от числата  $d(x,y)$ , където  $x$  и  $y$  са два произволни върха от графа. Напишете програмата **diam**, която намира диаметъра на графа.

**Вход**

Данните се въвеждат от стандартния вход. На първия ред е дадено числото  $n$  – броя на върховете в графа ( $0 < n < 1000$ ). Върховете на графа са номерирани с целите числа от 1 до  $n$ . Всеки от следващите  $n-1$  реда описва едно ребро от графа: първите две числа задават краищата на реброто, а третото число е дължината на реброто, цяло неотрицателно число по-малко от 1000.

**Изход**

На един ред на стандартния изход да се изведе търсеният диаметър.

**ПРИМЕР**

**Вход**

```
10
4 5 5
4 3 2
4 2 1
5 6 4
5 1 0
5 7 4
3 8 4
3 9 3
3 10 3
```

**Изход**

```
15
```

## Задача А2. МНОГОЪГЪЛНИЦИ

В равнината са дадени  $N$  различни точки. Никои три от тях не лежат на една права. Напишете програма **poly**, която намира най-малкото лице на изпъкнал  $K$ -ъгълник, чиито върхове са измежду дадените точки.

### Вход

На първия ред на стандартния вход са записани целите числа  $N$  и  $K$ . Следват  $N$  реда, всеки съдържащ двойка координати на поредната дадена точка. Между всеки две числа в редовете от входа има по един интервал.

### Изход

На един ред в стандартния изход вашата програма трябва да изведе едно цяло число, равно на цялата част от стойността на търсеното минимално лице. Ако не съществува нито един изпъкнал  $K$ -ъгълник от описания вид, вашата програма трябва да изведе число 0.

**Ограничения:**  $0 < N < 50$ ,  $0 < K < 11$ , координатите на дадените точки са цели неотрицателни числа, по-малки от 9999.

### ПРИМЕР

#### Вход

```
4 3
0 0
1 1
0 10
10 0
```

#### Изход

```
5
```

## Задача А3. ОЦВЕТЯВАНЕ

Дадена е правоъгълна таблица с  $n$  реда и  $k$  стълба. Две клетки наричаме съседни, когато имат обща страна. Напишете програма **color**, която намира по колко различни начина могат да се оцветят клетките на таблицата така, че всяка от тях да е черна или бяла и да има точно една съседна черна клетка.

### Вход

От един ред на стандартния вход се въвеждат стойностите на  $n$  и  $k$  ( $0 < n < 500$ ,  $0 < k < 32$ ).

### Изход

На един ред на стандартния изход програмата трябва да изведе търсения брой.

### ПРИМЕР

#### Вход

```
2 2
```

#### Изход

```
4
```

**XXIV НАЦИОНАЛНА ОЛИМПИАДА ПО ИНФОРМАТИКА**  
*Национален кръг, 18-20.04.2008 г.*

**Задача А4. ТАБЛИЦА**

Дадени са 12 различни естествени числа. Казваме, че сме ги наредили “правилно” в правоъгълна таблица 3 x 4 (три реда, четири колонки), ако сумата на числата във всеки от редовете е четно число, а сумата от числата във всяка от колонките се дели на 3. Това може би е реализуемо по много начини. Две разположения ще наричаме „еквивалентни”, ако едното се получава от другото чрез разместване на редове и/или на колонки. Очевидно е, че такива размени не влияят на исканите свойства, т. е., върху правилността на подредбата. Например, първата подредба, показана по-долу, удовлетворява условието за редовете и колонките и, следователно, е правилна. Втората е еквивалентна на първата – в нея ред 1 е предишният 2, ред 2 – предишният 1, а ред 3 си е на мястото. Подредба 3 пък е получена от 2 с размяна на колонки. Трите показани подредби са еквивалентни помежду си.

3	4	29	14
5	10	36	33
7	37	31	13

*Подредба 1*

5	10	36	33
3	4	29	14
7	37	31	13

*Подредба 2*

33	36	5	10
14	29	3	4
13	31	7	37

*Подредба 3*

Напишете програма **table**, която определя по колко нееквивалентни начина може да се направи правилна подредба от дадените числа.

**Вход**

От стандартния вход се въвеждат два реда. Всеки от тях съдържа по 12 естествени числа, разделени с интервал. Никое от числата няма повече от 50 цифри.

**Изход**

За всеки от входните редове запишете на стандартния изход съответен изходен ред с едно число, равно на броя на нееквивалентните правилни подредби на дванадесетте числа в таблица 3 x 4.

**ПРИМЕР**

**Вход**

```
6 3 1 7 9 11 12 10 2 4 5 8
6 12 18 24 30 36 42 48 54 59 60 61
```

**Изход**

```
40320
0
```

**Забележка:**

В 40% от тестовите данни числата не надминават 100.

## Задача А5. МАТРЬОШКИ

След като превзе всички възможни информатични върхове, Пешо Хакерът реши да опита ново предизвикателство. Регистрира се в руския сайт за онлайн състезания по криптиране и ключарство "ВерхШифр" и не след дълго се оказа в Москва на финален кръг.

По време на една от екскурзиите, които организаторите устройха, групата посети голям музей за матрьошки – традиционни руски играчки, които могат да се съдържат една в друга. Вътре имаше странен лабиринт. Стаите, както и целият лабиринт, бяха с квадратна форма, а стените им – успоредни на осите север-юг и изток-запад. Всяко помещение имаше по четири врати – по една на всяка стена. Входът беше в най-северозападната стая, а изходът – в най-югоизточната.

Руският национален отбор по шифри се беше постарал да направи нещата интересни за колегите си, като устрои състезание. Във всяка стая бяха поставили по една празна матрьошка за да подхожда на музея. Всеки гост трябваше да премине през лабиринта и да се опита да събере колкото се може повече от традиционните играчки, като поставя събраните до тук матрьошки в новата, която е решил да вземе. Разбира се, всеки участник сам преценяваше кога да вземе матрьошката в дадена стая (стига да е по-голяма от тези, които има) или просто да продължи към следващата стая. За капак геният на сейфовете – Метр Питричев – беше модифицирал вратите, така че да може да се преминава само в по-южна или по-източна посока.

Пешо, разбира се, не беше чак толкова добър с ключалките и не можа да се справи с ограничението на вратите. За това пък си написа програма, с която да извлече максимума от ситуацията. А вие можете ли? Напишете програма `matr`, която по зададена конфигурация на лабиринта намира с колко матрьошки е излязъл Хакерът.

### Вход

На стандартния вход ще получите едно цяло число  $N$  ( $1 < N < 1000$ ), последвано от  $N$  реда с по  $N$  числа (по едно за стая), всяко от които описва големината на матрьошката в съответното помещение. Всеки следващ ред от числа описва по-южна ивица от стаи, а всяка следваща колона – по-източна. Една матрьошка може да се сложи в друга (заедно с всичките матрьошки в нея), ако има по-малък размер от предишната. Размерите на играчките са неповтарящи се цели числа в интервала  $[1, N^2]$ .

### Изход

На стандартния изход трябва да изведете едно число – максималния брой матрьошки, които Пешо Хакерът може да събере, спазвайки правилата.

### ПРИМЕРИ

#### Вход

```
3
9 4 1
6 5 7
2 3 8
```

#### Изход

```
4
```

#### Вход

```
4
1 15 4 5
2 10 9 16
14 7 8 3
13 6 11 12
```

#### Изход

```
6
```

## Задача А6. СИМЕТРИЧНИ ПОЛИНОМИ

Дадено е цяло положително число  $N$ . Един полином на  $N$  променливи  $x_1, x_2, \dots, x_N$  се нарича симетричен, ако той не се променя при всевъзможните пермутации на номерата на променливите. Например при  $N=3$ , полиномите  $x_1+x_2+x_3$ ,  $x_1x_2+x_2x_3+x_1x_3$  и  $3x_1x_2+2x_2x_3+3x_1x_3+x_2x_3$  са симетрични. Полиномът  $x_1x_2$  е симетричен при  $N=2$ , но не е симетричен при  $N=3$ . Напишете програма **sympo1**, която определя дали даден полином е симетричен.

### Вход

На първия ред на стандартния вход е записан броят  $K$  на полиномите, които вашата програма трябва да провери. Следват описанията на  $K$  полинома. Данните за всеки от тях се състоят от ред, съдържащ стойностите на  $N$  и  $M$  – брой променливи на полинома и брой на едночлените, от който е съставен, следвани от  $M$  реда, всеки описващ поредния едночлен. Всеки от тези редове се състои от  $N+1$  цели числа – първото е коефициентът на едночлена, а следващите са степенните показатели на  $x_1, x_2, \dots, x_N$ . Числата във всеки ред от входа са разделени с интервали.

### Изход

На един ред в стандартния изход вашата програма трябва да изведе низ от 1 и 0 (без разделящи интервали), показващи дали съответните полиноми от входа са симетрични или не са такива (1 – симетричен, 0 – не е симетричен).

**Ограничения:**  $0 < K < 21$ ,  $0 < N < 20$ ,  $0 < M < 9999$ , коефициентите на едночлените са ненулеви цели числа между  $-99$  и  $99$ . Степенните показатели са неотрицателни цели числа, по-малки от 5.

### ПРИМЕР

#### Вход

```
2
3 1
1 1 1 0
3 4
3 1 1 0
2 0 1 1
3 1 0 1
1 0 1 1
```

#### Изход

```
01
```

**XXIV НАЦИОНАЛНА ОЛИМПИАДА ПО ИНФОРМАТИКА**  
*Национален кръг, 18-20.04.2008 г.*

**Задача С1. ИГРА**

Една игра, подобно на шахмата, се играе на квадратна дъска, разделена на  $N^2$  полета. Полетата са номерирани с числата от 1 до  $N^2$  по редове (Фиг. 1). Полетата са бели и черни, но за разлика от шахмата са оцветени произволно (Фиг. 2).

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

Фиг. 1

		■		
	■		■	
			■	
	■			
■				

Фиг. 2

Една пионка се движи само по бели полета, като от дадено поле може да отиде за един ход в друго поле, само ако двете полета се намират на една и съща хоризонтална или вертикална линия и между тях няма черни полета. Например при оцветяването на Фиг. 2 от поле 13 може да се отиде с един ход в полетата 8, 11, 12, 18, 23.

Напишете програмата **move**, която по дадени размери и оцветяване на дъската намира минималния брой ходове, с които пионката може да се придвижи от едно място до друго.

**Вход**

На първия ред на стандартния вход са дадени три числа  $N$ ,  $X$ ,  $Y$ , представляващи съответно размера на дъската и номерата на началното и крайното поле. На втория ред на стандартния вход са зададени броят  $B$  на черните полета и номерата на самите полета.

**Изход**

На един ред на стандартния изход програмата трябва да изведе търсения минимален брой ходове за придвижване по правилата на играта от полето  $X$  до полето  $Y$ . Ако не е възможно да се стигне от полето  $X$  до полето  $Y$ , да се изведе  $-1$ .

**Ограничения:**  $0 < N < 1000$

**ПРИМЕР**

**Вход**

5 2 4  
6 3 7 9 14 17 21

**Изход**

7

## Задача С2. ТРАНСФОРМАЦИИ

Нека  $a$  и  $b$  са естествени числа с равен брой цифри, всяка от които е различна от 0. С един ход в  $a$  може да променим  $k$  последователни цифри, като цифрите 1, 2, 3, 4, 5, 6, 7, 8 и 9 заменим съответно с цифрите 2, 3, 4, 5, 6, 7, 8, 9 и 1. Така например, ако  $a = 12349$  и  $k = 2$ , с един ход от  $a$  може да получим числата 23349, 13449, 12459 и 12351. Да се напише програма **trans**, която намира след колко хода най-малко от числото  $a$  може да се получи числото  $b$ .

### Вход

От първия ред на стандартния вход се въвежда  $k$  ( $1 < k < 10$ ). На следващите два реда се въвеждат числата  $a$  и  $b$  ( $a \neq b$ ), всяко от които има поне  $k$  и най-много 100 цифри.

### Изход

На един ред на стандартния изход програмата трябва да изведе търсения брой ходове. Ако от  $a$  не може да се получи  $b$ , програмата трябва да изведе 0.

### ПРИМЕР

#### Вход

```
2
123456789
123467791
```

#### Изход

```
2
```

### Задача С3. СТРАННИ ДУМИ

Много често Разсейко вместо да си учи уроците, се занимава с измисляне на разни странни думи. Последното му хрумване било да образува такива думи, при които щом една буква се среща веднъж, тя непременно се среща още веднъж в думата. По този начин, всяка буква от думата задължително се среща в нея четен брой пъти. Освен това, всички букви, които се намират между двете срещания трябва да образуват отново такава дума. Ако между двете срещания няма букви, то думата отговаря на условието. Например думите “abbacc”, “aaabbbbba”, “dddd” отговарят на условието на Разсейко, но думите “abcbac”, “aaa”, “aaabbbbab” – не.

От този момент заниманието на Разсейко било да проверява всяка дума дали отговаря на неговото условие. Нещо повече, той иска да знае коя е най-дългата странна дума във всеки урок, който започвал да учи. Проверянето му отнема толкова много време, че е наредил цяла върволица от двойки. Помогнете му като съставите програма **sword**, която прочита от клавиатурата текст, съставен само от думи, разделени с точно един интервал и извежда най-дългата странна дума в него. Думите в текста са съставени само от латински букви, като вашата програма не трябва да прави разлика между малките и главните букви.

#### Вход

От стандартният вход се въвежда текст съставен от думи, записани с малки и главни латински букви, като думите са отделени една от друга с точно един интервал. Сигурно е, че текстът съдържа поне една странна дума.

#### Изход

На единствения ред на стандартния изход се извежда точно една дума – най-дългата странна дума, която се среща в текста. Намерената дума се отпечатва само с малки букви. Ако има няколко думи, отговарящи на условието, които са с дължина, равна на най-дългата, се извежда тази от тях, която се среща първа в текста.

**Ограничения:** Текстът може да съдържа не повече от 20000 думи, всяка от които има не повече от 100 букви.

### ПРИМЕР

#### Вход

```
Abbacc dd ne da alabala soos sos abcbacb
```

#### Изход

```
Abbacc
```

**XXIV НАЦИОНАЛНА ОЛИМПИАДА ПО ИНФОРМАТИКА**  
*Национален кръг, 18-20.04.2008 г.*

**Задача С4. МНОГОЪГЪЛНИК**

Многоъгълник със страни, успоредни на координатните оси, е зададен посредством хоризонталните си страни. Напишете програма **poly**, която намира лицето на многоъгълника.

**Вход**

От първия ред на стандартния вход се въвежда цяло число  $N$  ( $2 \leq N \leq 1000$ ) – брой на хоризонталните страни на многоъгълника, а от следващите  $N$  реда – по три цели числа, които определят една хоризонтална страна – абсцисите на левия и десния край и ординатата им ( $-1000 \leq$  абсциси, ординати  $\leq 1000$ ).

**Изход**

На стандартния изход се извежда едно число – лицето на многоъгълника.

**ПРИМЕР**

**Вход**

```
3
0 10 10
10 20 0
0 20 20
```

**Изход**

```
300
```

## Задача С5. ВЕЗНА

Имаме везна и теглилки с маса  $1, 3, 9, 27, \dots, 3^{n-1}$  кг, като всяка теглилка е в единствен екземпляр. На лявото блюдо на везната е поставен предмет с маса  $m$  кг.

Напишете програма **scale**, която намира такова разпределение на теглилките върху двете блюда, че да се постигне баланс. Не е необходимо да се използват всички теглилки.

### Вход

От единствен ред на стандартния вход се въвеждат две цели числа –  $n$  ( $1 \leq n \leq 20$ ) и  $m$  ( $1 \leq m \leq 3^{n-1}$ ).

### Изход

На първия ред на стандартния изход програмата извежда числото  $m$ , последвано от масите на теглилките от лявото блюдо на везната в нарастващ ред, а на втория ред се извеждат масите на теглилките от дясното блюдо, също в нарастващ ред.

### ПРИМЕР

#### Вход

4 5

#### Изход

5 1 3  
9

## Задача С6. ТАНЦИ

На тренировка в танцов клуб “J&G” танцьор трябва да стигне до мястото, в което ще бъде партньорката му след определено време. Това място се намира на  $x$  крачки наляво и  $y$  крачки назад. Момчето използва поредица от два вида танцови стъпки. При първия вид танцьорът се придвижва две крачки наляво, а при втория – две крачки назад. Напишете програма **dance**, която определя броя на различните варианти, по-които танцьорът може да се придвижи до партньорката си.

### Вход

От първия ред на стандартния вход се въвеждат  $x$  и  $y$  ( $1 < x, y < 61$ ).

### Изход

Извежда се едно число, показващо броя на вариантите за различно придвижване на партньора до дамата. Ако не може да се стигне до нея по описания начин, да се изведе 0.

### ПРИМЕР

#### Вход

4 4

#### Изход

6

**XXIV НАЦИОНАЛНА ОЛИМПИАДА ПО ИНФОРМАТИКА**  
*Национален кръг, 18-20.04.2008 г.*

**Задача D1. ИЗРАЗ**

Даден е правилно съставен аритметичен израз, съставен само от знаците + и \*, и едноцифрени числа (+ е знак за събиране, \* е знак за умножение). Изразът е такъв, че знаците + и \* задължително се редуват във вида: +, \*, +, \*, ... Например, един такъв израз може да бъде следният:  $1+2*3+4*5$ . Общият брой на знаците + и \* не е по-голям от 10.

Напишете програма **expres**, която пресмята дадения израз (съгласно правилата на аритметиката).

**Вход:** Програмата прочита дадения израз като един низ от стандартния вход.

**Изход:** На един ред в стандартния изход програмата трябва да изведе резултата като едно цяло число.

**ПРИМЕР**

**Вход**

$1+2*3+4*5$

**Изход**

27

**Задача D2. СЪКРОВИЩЕ**

След мъчителна борба с бурното море корабокрушенци попаднали на брега на самотен остров. Там намерили необичайно послание, написано на лист хартия и поставено в бутилка. Най-отгоре на листа пишело, че на този остров има съкровище и ще го намери този, който успее да прочете посланието. По-надолу върху листа измежду  $N$  реда със знаци се криело и самото послание. Знаците били главни и малки букви от английската азбука. От всеки отделен ред трябвало да се открие по един знак от посланието. За целта се изисквало да се намери най-малкият знак, който не се среща в реда и е по-голям от най-малкият знак в реда (приемаме, че една буква е по-голяма от друга, ако тя е подредена след нея в английската азбука и освен това, считаме че всички главни букви са подредени в азбучен ред в началото на азбуката и след тях следват всички малки букви, подредени също в азбучен ред). Ако редът съдържа всичките английски букви – главни и малки, търсеният знак е точка ('.'). Напишете програма **treasure**, която разчита посланието, намерено от корабокрушенците.

**Вход**

От първия ред на стандартния вход се въвежда  $N$  ( $1 \leq N \leq 10000$ ). От следващите  $N$  реда се въвеждат до 10000 знака в ред.

**Изход**

На стандартния изход да се изведе посланието съдържащо  $N$  на брой знака, което ни показва къде е съкровището.

**ПРИМЕР****Вход**

7  
SPhA  
lgfhpqv  
efys  
znSZww  
pnqo  
cdbay  
psxd

**Изход**

BigTree

*Пояснение:* Най-малката буква в първия ред е A. Следващата по големина, която не се среща в реда е B. Във втория ред най-малката буква е f – следващата по големина, която не се среща е i, защото в реда се срещат буквите g и h.

**Задача D3. ЦИФРА**

Иван започнал да въвежда в един файл едно след друго числата 1, 2, 3, ..., 9, 10, 11, ...  
Напишете програма **digit**, която определя коя е  $N$ -тата въведена цифра.

**Вход**

От стандартния вход се въвежда  $N$  ( $1 \leq N \leq 10000$ ).

**Изход**

На стандартния изход да се изведе търсената цифра.

**ПРИМЕРИ****Вход**

1

**Вход**

11

**Вход**

15

**Изход**

1

**Изход**

0

**Изход**

2

**XXIV НАЦИОНАЛНА ОЛИМПИАДА ПО ИНФОРМАТИКА**  
*Национален кръг, 18-20.04.2008 г.*

**Задача E1. РОМБОВЕ**

Да разгледаме фигурата, образувана от три, вписани един в друг ромба със страни от звездички (\*):

```
*
***
*****
***  ***
*****
***
*
```

Страната на най-вътрешния ромб е от две звездички, а страната на всеки следващ ромб е с едно по-голяма. Напишете програма **fig**, която по зададено  $N$  извежда аналогична фигура от  $N$  вписани ромба.

**Вход**

От стандартния вход се въвежда броят  $N$  на вписаните ромбове.

**Изход**

На стандартния изход се извежда фигурата. Не трябва да има празни редове преди реда, съдържащ първата звездичка. Не трябва да има интервали отляво на първата звездичка в този от редовете, който съдържа звездичка в най-лява позиция.

**Ограничения:**  $0 < N < 15$

**ПРИМЕР**

**Вход**

3

**Изход**

```
*
***
*****
***  ***
*****
***
*
```

## Задача Е2. ОБИКОЛКА

Иванчо тръгнал с велосипед на обиколка из България. Всеки ден си записвал колко часа и минути е пътувал и какво разстояние в километри е изминал. Напишете програма **cycling**, която да изведе общото време, през което Иванчо е карал велосипеда си и номера на деня, когато е карал с най-голяма средна скорост. Средната скорост за деня е равна на разстоянието в метри, което е изминато през деня, разделено на минутите, през които е пътувано.

### Вход

Вашата програма прочита от първия ред на стандартния вход броя  $N$  на дните. Следват  $N$  реда, всеки от които съдържа по три цели числа, разделени с интервали – часовете и минутите на пътуването, и изминатото разстояние в километри.

### Изход

На един ред на стандартния изход вашата програма трябва да изведе четири цели числа, разделени с по един интервал: общото време на пътуването във вид на дни, часове и минути, и номера на деня, когато е пътувал с най-голяма средна скорост.

**Ограничения:**  $0 < N < 10$

## ПРИМЕР

### Вход

```
3
20 15 400
10 30 320
5 0 40
```

### Изход

```
1 11 45 2
```

## Задача Е3. НЕСЪКРАТИМА ДРОБ

Напишете програма **frac**, която съкращава обикновени дроби.

### Вход

От първия ред на стандартния вход се въвеждат две цели числа  $a$  и  $b$ , които са съответно числител и знаменател на обикновена дроб ( $0 < a < b < 1000$ ).

### Изход

На един ред в стандартния изход да се изведат числителя и знаменателя на дробта, след извършването на всички възможни съкращения.

## ПРИМЕРИ

### Вход

```
4 10
```

### Изход

```
2 5
```

### Вход

```
22 35
```

### Изход

```
22 35
```

### Вход

```
315 385
```

### Изход

```
9 11
```