

БУРГАСКИ СВОБОДЕН УНИВЕРСИТЕТ
Междунниверситетско състезание по програмиране
24 март 2002 г.

Задача 1. Двойки числа

Напишете програма, която намира двойка естествени числа n_1 и n_2 , по-малки от 10000 и удовлетворяващи следните три условия:

1. Частното на двете числа $\frac{n_1}{n_2}$ е предварително зададено цяло число N , ($1 < N < 10000$);
2. Сумата от цифрите на първото число е равна на сумата от цифрите на второто число;
3. Разликата между двете числа $n_1 - n_2$ е минимална, т.е. измежду всички двойки числа, удовлетворяващи 1 и 2 търсим двойката най-близки числа.

Име на програмата – div<номер на отбор>.exe:

Вход – текстов файл div.in:

Състои се от няколко (най-много 10) реда, всеки съдържащ по една стойност на числото N . На края на файла има още един ред, съдържащ само числото 0.

Изход – текстов файл div<номер на отбор>.out:

Трябва да съдържа толкова реда, колкото са редовете във входния файл, съдържащи стойности за N , като във всеки от редове в изходния файл трябва се запише намерената стойност на разликата от т. 3 за съответната входна данна или символ * (звезда), ако задачата няма решение.

Примерен вход:

```
99
1244
8
0
```

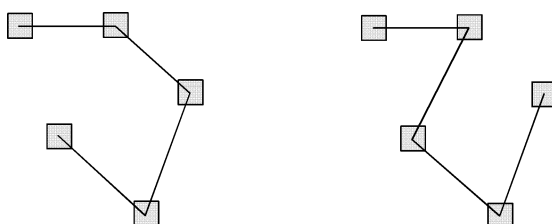
Примерен изход:

```
9702
*
63
```

БУРГАСКИ СВОБОДЕН УНИВЕРСИТЕТ
Междуниверситетско състезание по програмиране
24 март 2002 г.

Задача 2. Компютърна мрежа

Да се напише програма за свързване на компютри в локална мрежа. Прави се линейна мрежа, т.е. компютрите са свързани последователно (коаксиална мрежа), като стремежът е общата дължина на проводника да бъде възможно най-малка. Компютрите са разположени в една равнина и предполагаме, че е възможно прекарване на кабел по права линия между всеки два компютъра. Дадени са координатите на всеки компютър. Да се намери такова свързване, че общата дължина на кабелите е минимално възможната. Броят на компютрите е по-малък от 10.



Име на програмата – lan<номер на отбор>.exe:

Вход – текстов файл lan.in

Съдържа няколко (най-много 10) тестови примери. Данните за всеки от примерите са записани на няколко последователни реда във файла. Първият от редовете за всеки от тестовете съдържа числото N – брой компютри, като следват още N реда, всеки съдържащ координатите им – двойка цели числа (със стойности между 0 и 1000), разделени с една шпация. Появата на числото 0, като брой компютри означава край на входния файл.

Изход – текстов файл lan<номер на отбор>.out

Трябва да съдържа толкова реда, колкото са тестовите примери във входния файл. За всеки пример трябва да се изведе на съответния ред в изходния файл намерената минималната дължина на кабела като число с десетична точка с точно 3 цифри в дробната му част, като се приложи закръгляне чрез отрязване на следващите цифри.

Примерен вход:

```
5
11 27
84 99
142 81
88 30
95 38
3
101 102
111 112
108 109
0
```

Примерен изход:

```
210.401
14.142
```

БУРГАСКИ СВОБОДЕН УНИВЕРСИТЕТ
Междуниверситетско състезание по програмиране
24 март 2002 г.

Задача 3. Политическа сила

Всички важни решения в Европейския съюз се вземат чрез гласуване с „да“ или „не“ на дадено предложение, като участващите в съюза държави имат различен брой гласове: Франция, Германия, Италия и Англия – по 10 гласа, Испания – 8, Белгия, Холандия, Португалия и Гърция – 5, Австрия и Швеция – 4, Дания, Ирландия и Финландия – 3, Люксембург – 2 гласа. Едно предложение се приема, ако за него са гласували с „да“ 62 от общо 87 гласа.

Мярка за политическата сила на една държава в съюза е индексът на Банзаф. Нека p е дадена държава. Обща политическа сила $O(p)$ на p се нарича броят на всевъзможните коалиции C от държави, изпълняващи следните 3 условия:

1. p е член на коалиция C ;
2. C е печеливша коалиция (събира необходимите брой гласове за приемане на дадено предложение);
3. Когато p излезе от C , получената коалиция е губеща (не е печеливша).

Индекс на Банзаф $B(p)$ за държавата p се пресмята по формулата: $B(p) = O(p)/s$, където s е сумата от общата политическа сила на всички участници в съюза.

Да се напише програма `uni<номер на отбор>.exe` за пресмятане индекса на Банзаф.

Вход – текстов файл `uni.in`

Съдържа няколко (най-много 10) тестови примери. Данните за всеки от примерите са записани на два последователни реда във файла. Първият ред за всеки от примерите съдържа две цели числа, разделени с една шпация – броят N на държавите в съюза ($1 < N < 32$) и необходимият брой гласове V за вземане на решение. На следващия ред има N цели числа (разделени с по една шпация), които са гласовете на участниците в съюза. Файлът завършва с ред, съдържащ числото 0.

Изход – текстов файл `uni<номер на отбор>.out`

За всеки пример трябва да се изведат N цели числа, по едно на ред, всяко равно на индекса на Банзаф, изразен в проценти за поредния участник в съюза (закръглен до най-близкото цяло число). Между изходните данни за отделните примери трябва да се остави по един празен ред.

Примерен вход:

```
15 62
10 10 10 10 8 5 5 5 5 4 4 3 3 3 2
2 50
49 51
0
```

Примерен изход:

```
11
11
11
11
9
6
6
6
6
5
5
4
4
4
4
2
```

Време за работа на програмата 5 сек.

```
0
100
```

БУРГАСКИ СВОБОДЕН УНИВЕРСИТЕТ
Междууниверситетско състезание по програмиране
24 март 2002 г.

Задача 4. Покрития

Напишете програма, която намира по колко различни начина един правоъгълник с размери $n \times m$, ($0 < m < 12$, $0 < n < 8$, m и n са цели числа) може да се покрие изцяло с еднакви по-малки неразличими правоъгълници с размери 1×2 , като по-малките правоъгълници не излизат извън големия, не се застъпват един с друг и страните им се поставят успоредни на страните на големия правоъгълник.

Име на програмата – cov<номер на отбор>.exe

Вход – текстов файл cov.in

Състои се от няколко (най-много 10) реда, всеки съдържащ стойности на двете числа m и n , разделени с една шпация. На края на файла има още един ред, съдържащ само числото 0.

Изход – текстов файл cov<номер на отбор>.out

Трябва да съдържа толкова реда, колкото са редовете във входния файл, съдържащи стойности на m и n , като във всеки от тези редове трябва се запише намереното решение на задачата за съответните входни данни.

Примерен вход:

```
1 1
2 3
0
```

Примерен изход:

```
0
3
```

БУРГАСКИ СВОБОДЕН УНИВЕРСИТЕТ
Междууниверситетско състезание по програмиране
24 март 2002 г.

Задача 5. Броене

Густаво знае да брои, но сега той се учи да пише числата. Като много добър ученик той е научил 1, 2, 3 и 4. Но той не осъзнава, че 4 е различно от 1. Въпреки това, той се забавлява с една игра, в която съставя числа от тези цифри и смята сбора на цифрите. Например:

$$132 = 1 + 3 + 2 = 6$$

$$112314 = 1 + 1 + 2 + 3 + 1 + 1 = 9 \text{ (запомнете, че Густаво мисли, че } 4 = 1 \text{)}$$

След като създава много числа по този начин, сега той иска да разбере колко числа може да направи така, че тяхната сума да е равна на N . Например за $N = 2$ той може да напише 5 числа: 11, 14, 41, 44 и 2. Понеже бил зает с учене, той ви моли за помощ.

Име на програмата – `count<номер на отбор>.exe`:

Вход – текстов файл `count.in`

Съдържа няколко стойности на числото N .

Изход – текстов файл `count<номер на отбор>.out`

Съдържа по едно число на всеки ред: броя числа, чиято сума от цифрите е съответното N .

Примерен вход:

2

3

Примерен изход:

5

13