

## Задача А. Редактор

Имя входного файла: INPUT.TXT  
Имя выходного файла: OUTPUT.TXT  
Максимальное время работы на каждом тесте: 10 секунд

Компания *Macrohard* выпустила новую версию своего редактора *Nottoobad*, который понимает некоторые голосовые команды. К сожалению, этих команд всего две – "повторить последнее слово" и "стереть последний символ". Причем при исполнении команды "повторить последнее слово" редактор автоматически вставляет пробел, который разделяет слова.

Однако компания утверждает, что с помощью этого редактора можно набирать текст, нажимая клавиши на клавиатуре гораздо реже. Например, чтобы набрать фразу "this thin thing" достаточно нажать на клавиши на клавиатуре всего 6 раз:

Действие	Нажатий	Содержимое документа
Набрать "this"	4	this
Сказать "повторить последнее слово"	0	this this
Сказать "стереть последний символ"	0	this thi
Набрать "n"	1	this thin
Сказать "повторить последнее слово"	0	this thin thin
Набрать "g"	1	this thin thing

Чтобы повысить популярность своего продукта, компания решила провести конкурс, победителем которого станет тот, кто сможет набрать заданный набор слов в редакторе за наименьшее количество нажатий на клавиши. Причем первое слово зафиксировано, а остальные могут быть набраны в произвольном порядке. То есть, если надо набрать слова "apple", "plum" и "apricote", то первым надо набрать "apple", а слова "plum" и "apricote" можно поменять местами.

Поскольку Вы собираетесь участвовать в конкурсе, и у Вас есть знакомый в компании, который сообщил Вам по секрету набор слов, которые надо будет набрать, то неплохо бы написать программу, которая найдет порядок набора слов, при котором количество нажатий на клавиши будет минимальным.

### Формат входных данных

На первой строке входного файла находится число  $N$  ( $1 \leq N \leq 100$ ) – количество слов, которые предстоит набрать. Следующие  $N$  строк содержат слова – последовательности маленьких латинских букв, не длинее 100 символов. Помните, что первое слово необходимо набрать первым!

### Формат выходных данных

Выведите в выходной файл на первой строке число  $X$  - минимальное количество нажатий на клавиши, которое придется совершить, чтобы набрать все указанные слова в редакторе *Nottoobad*. На следующих  $N$  строках выведите слова в том порядке, в котором их следует набирать для достижения этого количества нажатий. Если решений несколько, выведите любое из них.

### Примеры

INPUT.TXT	OUTPUT.TXT
3 this thin thing	6 this thin thing
4 popcorn apple apricote plum	21 popcorn plum apricote apple
2 hello hello	5 hello hello

## Задача В. Марсианские факториалы

Имя входного файла: INPUT.TXT  
Имя выходного файла: OUTPUT.TXT  
Максимальное время работы на каждом тесте: 10 секунд

В 3141 году очередная экспедиция на Марс обнаружила в одной из пещер таинственные знаки. Они однозначно доказывали существование на Марсе разумных существ. Однако смысл этих таинственных знаков долгое время оставался неизвестным. Недавно один из ученых, профессор Очень-Умный, заметил один интересный факт: всего в надписях, составленных из этих знаков, встречается ровно  $K$  различных символов. Более того, все надписи заканчиваются на длинную последовательность одних и тех же символов.

Вывод, который сделал из своих наблюдений профессор, потряс всех ученых Земли. Он предположил, что эти надписи являются записями факториалов различных натуральных чисел в системе счисления с основанием  $K$ . А символы в конце – это конечно же нули – ведь, как известно, факториалы больших чисел заканчиваются большим количеством нулей. Например, в нашей десятичной системе счисления факториалы заканчиваются на нули начиная с  $5! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 = 120$ . А у числа  $100!$  в конце следует 24 нуля в десятичной системе счисления и 48 нулей в системе счисления с основанием 6 – так что у предположения профессора есть разумные основания!

Теперь ученым срочно нужна программа, которая по заданным числам  $N$  и  $K$  найдет количество нулей в конце записи в системе счисления с основанием  $K$  числа  $N! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot (N-1) \cdot N$ , чтобы они могли проверить свою гипотезу. Вам придется написать им такую программу!

### Формат входных данных

На первой строке входного файла находятся числа  $N$  и  $K$ , разделенные пробелом. ( $1 \leq N \leq 10^9$ ,  $2 \leq K \leq 1000$ ).

### Формат выходных данных

Выведите в выходной файл число  $X$  – количество нулей в конце записи числа  $N!$  в системе счисления с основанием  $K$ .

### Примеры

INPUT.TXT	OUTPUT.TXT
5 10	1
100 10	24
100 6	48
3 10	0

## Задача С. Фонтан

Имя входного файла: INPUT.TXT  
Имя выходного файла: OUTPUT.TXT  
Максимальное время работы на каждом тесте: 10 секунд

Администрация одного института решила построить в холле фонтан. По плану администрации, фонтан должен иметь форму круга с максимально возможным радиусом. Дизайнеру сообщили, что холл института имеет вид прямоугольника, размером  $X \times Y$  метров. Однако когда дизайнер стал выбирать место для фонтана, он столкнулся с серьезной проблемой: в холле института обнаружилось  $N$  круглых колонн, снести которые не представляется возможным.

Таким образом, у него появилась проблема: где следует поместить фонтан, чтобы он имел максимально возможный радиус и не имел ненулевого по площади пересечения с колоннами. Вам предстоит помочь ему в решении этой нелегкой задачи.

### Формат входных данных

На первой строке входного файла находятся вещественные числа  $X$  и  $Y$ ,  $1 \leq X, Y \leq 10^4$ . Будем считать, что прямоугольник холла расположен на координатной сетке так, что его углы имеют координаты  $(0,0)$ ,  $(X,0)$ ,  $(X,Y)$  и  $(0,Y)$ .

На второй строке находится число  $N$  ( $0 \leq N \leq 10$ ) – количество колонн. Следующие  $N$  строк содержат параметры колонн –  $i$ -я строка содержит три вещественных числа  $X_i$ ,  $Y_i$  и  $R_i$  – координаты центра и радиус  $i$ -й колонны ( $R_i \leq X_i \leq X - R_i$ ,  $R_i \leq Y_i \leq Y - R_i$ ,  $0.1 \leq R_i \leq \min\left(\frac{X}{2}, \frac{Y}{2}\right)$ ; для любых  $i \neq j$   $\sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2} \geq R_i + R_j$ ). Все числа во входном файле разделены пробелами.

### Формат выходных данных

Выведите в выходной файл три вещественных числа:  $X_F$ ,  $Y_F$  и  $R_F$  – координаты центра и радиус фонтана. Фонтан должен быть полностью расположен внутри холла (допускается касание стен) и не иметь ненулевого пересечения ни с одной из колонн (допускается касание). Радиус фонтана должен быть максимален. Разделяйте числа пробелами и/или переводами строки. Если решений несколько, выведите любое из них.

### Примечание о точности вычислений

При проверке результата работы Вашей программы нарушение условий оптимальности и отсутствия пересечения со стенами и колоннами менее чем на  $10^{-2}$  будет игнорироваться.

### Примеры

INPUT.TXT	OUTPUT.TXT
10 20 0	5.000 5.000 5.000
20 20 4 2 2 2 18 2 2 2 18 2 18 18 2	10.000 10.000 9.314
20 20 4 2 2 2 18 2 2 3 17 2 16 16 4	9.510 7.054 7.053

### Задача D. Детский праздник

Имя входного файла: INPUT.TXT  
Имя выходного файла: OUTPUT.TXT  
Максимальное время работы на каждом тесте: 10 секунд

Организаторы детского праздника планируют надуть для него  $M$  воздушных шариков. С этой целью они пригласили  $N$  добровольных помощников,  $i$ -й среди которых надувает шарик за  $T_i$  минут, однако каждый раз после надувания  $Z_i$  шариков устает и отдыхает  $Y_i$  минут. Теперь организаторы праздника хотят узнать, через какое время будут надуты все шарики при наиболее оптимальной работе помощников, и сколько шариков надует каждый из них. (Если помощник надул шарик, и должен отдохнуть, но больше шариков ему надувать не придется, то считается, что он закончил работу сразу после окончания надувания последнего шарика, а не после отдыха).

#### Формат входных данных

На первой строке входного файла находятся числа  $M$  и  $N$  ( $0 \leq M \leq 1000$ ,  $1 \leq N \leq 20$ ). Следующие  $N$  строк содержат по три целых числа -  $T_i$ ,  $Z_i$  и  $Y_i$  соответственно ( $1 \leq T_i, Y_i \leq 100$ ,  $1 \leq Z_i \leq 1000$ ).

#### Формат выходных данных

Выведите в выходной файл на первой строке число  $T$  - время, за которое будут надуты все шарики. На второй строке выведите  $N$  чисел – сколько шариков надует каждый из приглашенных помощников. Разделяйте числа пробелами. Если распределений шариков несколько, выведите любое из них.

#### Примеры

INPUT.TXT	OUTPUT.TXT
10 3 1 2 3 3 10 3 2 4 3	8 4 2 4
1 3 1 1 100 2 1 100 3 1 100	1 1 0 0

## Задача Е. Симпатичные узоры

Имя входного файла: INPUT.TXT  
Имя выходного файла: OUTPUT.TXT  
Максимальное время работы на каждом тесте: 10 секунд

Компания *BrokenTiles* планирует заняться выкладыванием во дворах у состоятельных клиентов узор из черных и белых плиток, каждая из которых имеет размер  $1 \times 1$  метр. Известно, что дворы всех состоятельных людей имеют наиболее модную на сегодня форму прямоугольника  $M \times N$  метров.

Однако при составлении финансового плана у директора этой организации появилось целых две серьезных проблемы: во первых, каждый новый клиент очевидно захочет, чтобы узор, выложенный у него во дворе, отличался от узоров всех остальных клиентов этой фирмы, а во вторых, этот узор должен быть симпатичным.

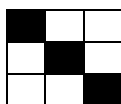


Рисунок 1

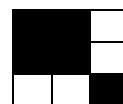
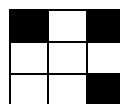
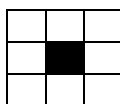
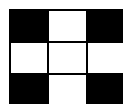
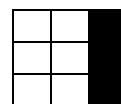


Рисунок 2



Как показало исследование, узор является симпатичным, если в нем нигде не встречается квадрата  $2 \times 2$  метра, полностью покрытого плитками одного цвета. На рисунке 1 показаны примеры различных симпатичных узоров, а на рисунке 2 – несимпатичных.

Для составления финансового плана директору необходимо узнать, сколько клиентов он сможет обслужить, прежде чем симпатичные узоры данного размера закончатся. Помогите ему!

### Формат входных данных

На первой строке входного файла находятся два положительных целых числа, разделенные пробелом -  $M$  и  $N$  ( $1 \leq M \cdot N \leq 30$ ).

### Формат выходных данных

Выведите в выходной файл единственное число – количество различных симпатичных узоров, которые можно выложить во дворе размера  $M \times N$ . Узоры, получающиеся друг из друга сдвигом, поворотом или отражением считаются различными.

### Примеры

INPUT . TXT	OUTPUT . TXT
2 2	14
3 3	322

## Задача F. Кубики

Имя входного файла: INPUT.TXT  
Имя выходного файла: OUTPUT.TXT  
Максимальное время работы на каждом тесте: 10 секунд

Родители подарили Пете набор детских кубиков. Поскольку Петя скоро пойдет в школу, они купили ему кубики с буквами. На каждой из шести граней каждого кубика написана буква.

Теперь Петя хочет похвастаться перед старшей сестрой, что научился читать. Для этого он хочет сложить из кубиков ее имя. Но это оказалось довольно сложно сделать – ведь разные буквы могут находиться на одном и том же кубике и тогда Петя не сможет использовать обе буквы в слове. Правда одна и та же буква может встречаться на разных кубиках. Помогите Пете!

Дан набор кубиков и имя сестры. Выясните, можно ли выложить ее имя с помощью этих кубиков и если да, то в каком порядке следует выложить кубики.

### Формат входных данных

На первой строке входного файла находится число  $N$  ( $1 \leq N \leq 100$ ) - количество кубиков в наборе у Пети. На второй строке записано имя Петиной сестры – слово, состоящее только из больших латинских букв, не длиннее 100 символов. Следующие  $N$  строк содержат по 6 букв (только большие латинские буквы), которые написаны на соответствующем кубике.

### Формат выходных данных

На первой строке выходного файла выведите "YES" если выложить имя Петиной сестры данными кубиками можно, "NO" в противном случае.

Если ответ "YES", на второй строке выведите  $M$  различных чисел из диапазона  $1..N$ , где  $M$  - количество букв в имени Петиной сестры.  $i$ -е число должно быть номером кубика, который следует положить на  $i$ -е место при составлении имени Петиной сестры. Кубики нумеруются с 1, в том порядке, в котором они заданы во входном файле. Если решений несколько, выведите любое. Разделяйте числа пробелами.

### Примеры

INPUT.TXT	OUTPUT.TXT
4 ANN ANNNNN BCDEFG HIJKLM NOPQRS	NO
5 HELEN ABCDEF GHIJKL MNOPQL STUVWN EIUOZK	YES 2 1 3 5 4

### Задача G. Волшебная последовательность

Имя входного файла: INPUT.TXT  
Имя выходного файла: OUTPUT.TXT  
Максимальное время работы на каждом тесте: 10 секунд

Недавно Петя научился считать. Он тут же заметил, что число 2 обладает замечательным свойством -  $2 + 2 = 2 \times 2$ . Его старший брат Ваня тут же объяснил ему, что дело не в двойке.

"Дело в том, что последовательность 2, 2 – волшебная," – сказал Пете Ваня. – "Волшебная последовательность – это такая последовательность натуральных чисел, что сумма ее членов равна их произведению. Например последовательность 1, 2, 3 – тоже волшебная."

Петя тут же сложил 1, 2 и 3, потом перемножил их и обрадовался.

Теперь Петя хочет найти более длинные волшебные последовательности. Помогите ему!

#### Формат входных данных

На первой строке входного файла содержится целое число  $N$  ( $2 \leq N \leq 100$ ).

#### Формат выходных данных

Выведите в выходной файл любую волшебную последовательность длины  $N$ . Разделяйте числа пробелами. Члены последовательности не должны превышать  $10^9$ . Если такой волшебной последовательности длины  $N$  не существует, выведите в выходной файл единственное число – "-1".

#### Примеры

INPUT . TXT	OUTPUT . TXT
2	2 2
3	1 2 3

## Задача Н. Похожие матрицы

Имя входного файла: INPUT.TXT  
Имя выходного файла: OUTPUT.TXT  
Максимальное время работы на каждом тесте: 10 секунд

Рассмотрим таблицу, состоящую из  $N$  строк и  $M$  столбцов. Если в каждой ячейке такой таблицы стоит целое число, назовем такую таблицу *целочисленной матрицей*. Скажем, что эта матрица *кратна числу  $p$* , если все числа в ее ячейках кратны  $p$ .

Рассмотрим теперь суммы элементов матрицы по строкам и столбцам соответственно. Обозначим сумму чисел  $i$ -й строки за  $H_i$ , а сумму чисел  $j$ -го столбца за  $V_j$ . Упорядоченный набор чисел  $(H_1, H_2, \dots, H_N, V_1, V_2, \dots, V_M)$  назовем *профилем матрицы*. Скажем, что матрица *почти кратна  $p$* , если все числа, входящие в ее профиль, кратны  $p$ .

Почти кратная 5 матрица и ее профиль изображены на рисунке 1.

6	7	2	15
2	2	31	35
7	1	7	15
15	10	40	

Рисунок 1

Если две матрицы  $A$  и  $B$  имеют одинаковый размер, причем элемент, стоящий на пересечении  $i$ -й строки и  $j$ -го столбца в матрице  $A$  отличается от соответствующего элемента матрицы  $B$  не более чем на  $p$ , скажем, что  $A$  отличается от  $B$  не более чем на  $p$ . Скажем, что матрица  $B$  *похожа на матрицу  $A$  относительно числа  $p$* , если

1.  $B$  отличается от  $A$  не более чем на  $p$
2. Профили  $B$  и  $A$  совпадают.

А	В	С																											
<table><tr><td>6</td><td>7</td><td>2</td></tr><tr><td>2</td><td>2</td><td>31</td></tr><tr><td>7</td><td>1</td><td>7</td></tr></table>	6	7	2	2	2	31	7	1	7	<table><tr><td>5</td><td>5</td><td>5</td></tr><tr><td>0</td><td>5</td><td>30</td></tr><tr><td>10</td><td>0</td><td>5</td></tr></table>	5	5	5	0	5	30	10	0	5	<table><tr><td>5</td><td>5</td><td>5</td></tr><tr><td>0</td><td>0</td><td>35</td></tr><tr><td>10</td><td>5</td><td>0</td></tr></table>	5	5	5	0	0	35	10	5	0
6	7	2																											
2	2	31																											
7	1	7																											
5	5	5																											
0	5	30																											
10	0	5																											
5	5	5																											
0	0	35																											
10	5	0																											
почти кратна 5	кратна 5, похожа на А	кратна 5, но непохожа на А																											

Рисунок 2

На рисунке 2 изображены две похожие относительно числа 5 матрицы, первая из них почти кратна 5, а вторая кратна 5. Третья матрица на рисунке 2 тоже кратна 5, но непохожа на первую (хотя похожа на вторую).

Дано число  $p$  и почти кратная  $p$  матрица  $A$ . Ваша задача – найти такую матрицу  $B$ , чтобы она была кратна  $p$  и похожа на  $A$  относительно  $p$ .

### Формат входных данных

На первой строке входного файла находятся целые числа  $p$  ( $1 \leq p \leq 10$ ),  $N$  и  $M$  ( $1 \leq N, M \leq 30$ ). Следующие  $N$  строк содержат по  $M$  целых неотрицательных чисел, не превышающих 1000, которые являются элементами исходной матрицы  $A$ .

### Формат выходных данных

Выведите в выходной файл матрицу  $B$  по строкам – сначала  $M$  элементов первой строки, затем  $M$  элементов второй, и т. д. Разделяйте числа пробелами и/или переводами строк. Заботиться о красивом форматировании таблицы не надо. Если искомой матрицы не существует, выведите единственное число – "-1". Если решений несколько, выведите любое из них.

### Пример

INPUT . TXT	OUTPUT . TXT
5 3 3 6 7 2 2 2 31 7 1 7	5 5 5 0 5 30 10 0 5



