

## Задача А. Игра

Имя входного файла:	<code>game.in</code>
Имя выходного файла:	<code>game.out</code>
Ограничение по времени:	2 секунда
Ограничение по памяти:	256 мегабайт

Камень, ножницы, бумага — популярная детская игра на руках, известная во многих странах мира. Иногда используется как методика случайного выбора персоны для какой-либо цели

---

wikipedia

Всем известны правила игры «Камень, ножницы, бумага» — играющие одновременно показывают при помощи руки один из трёх знаков: камень, ножницы или бумагу.

При этом, победитель определяется по следующим правилам:

- Камень побеждает ножницы.
- Ножницы побеждают бумагу.
- Бумага побеждает камень.

Ничья засчитывается в ситуации, когда одновременно хотя бы один игрок показал «камень», хотя бы один игрок показал «бумагу» и хотя бы один игрок показал «ножницы».

Леонарду и его друзьям надоело играть в обычную версию этой игры, поэтому они решили придумать свою собственную. В их версии игры знаков будет не три, а целых  $n$ . Для каждой пары знаков они заранее определили, какой из знаков будет бить другой знак. Победившим объявляется игрок, знак которого не проигрывает ни одному из других знаков и выигрывает хотя бы у одного другого знака. Если таких людей несколько, то они все объявляются победителями. Если таких игроков нет, то объявляется ничья.

Друзья решили сыграть в новую версию игры и позвали своих друзей, в итоге получилось  $m$  человек. Каждый из них показал один из  $n$  знаков. Теперь перед ними возникла новая проблема — определить, кто выиграл в этой игре. Так как они очень устали, они попросили вас помочь им.

### Формат входного файла

В первой строке входного файла даны два целых числа  $n$  и  $m$  — количество знаков и количество людей в игре соответственно ( $1 \leq n, m \leq 100$ ). В следующих  $n$  строках задана матрица правил этой игры. На  $j$ -ой позиции  $i$ -ой строки находится число 1, если  $i$ -ый знак побеждает  $j$ -ый знак, и 0 иначе. Гарантируется, что на  $i$ -ой позиции в  $i$ -ой строки находится число 0. В последней строке входного файла задано  $m$  целых чисел  $a_i$  ( $1 \leq a_i \leq n$ ) — номер знака, который выбрал  $i$ -ый друг.

### Формат выходного файла

В единственной строке выходного файла выведите номера всех выигравших друзей в возрастающем порядке. В случае ничьей выведите единственное число -1.

## Примеры

game.in	game.out
3 2 0 1 0 0 0 1 1 0 0 1 2	1
3 2 0 1 0 0 0 1 1 0 0 2 2	-1
3 3 0 1 0 0 0 1 1 0 0 1 2 3	-1

## Задача В. Записка

Имя входного файла: `letter.in`  
Имя выходного файла: `letter.out`  
Ограничение по времени: 2 секунда  
Ограничение по памяти: 256 мегабайт

Леонард и Пенни поругались. Леонард пытается извиниться, поэтому он написал записку, в которой он просит прощения. Но просто отдать записку — не интересно, поэтому Леонард её зашифровал. Чтобы по зашифрованному тексту получить исходный требуется  $k$  раз применить к тексту операцию циклического сдвига вправо. К сожалению, у Пенни нет компьютера, поэтому она попросила вас помочь ей расшифровать записку Леонарда.

Циклическим сдвигом строки  $s$  вправо называется строка  $t$  такая, что  $t_1 = s_{|s|}, t_2 = s_1, \dots, t_{|s|} = s_{|s|-1}$ , где  $|s|$  — длина строки  $s$ , а  $s_i$  —  $i$ -ый символ строки.

Рассмотрим все циклические сдвиги строки `abcde`

1. `eabcd`
2. `deabc`
3. `cdeab`
4. `bcdea`
5. `abcde`

### Формат входного файла

В первой строке входного файла задано два целых числа  $n$  и  $k$  ( $1 \leq k \leq n \leq 100000$ ) — длина зашифрованной строки и количество циклических сдвигов соответственно. Во второй строке входного файла находится текст записки. Текст состоит из строчных и прописных букв латинского алфавита, пробелов ( ), запятых (,) и точек (.).

### Формат выходного файла

В единственной строке выходного файла выведите расшифрованный текст записки.

### Примеры

<code>letter.in</code>
<code>56 5</code> <code>sorry, I am idiot. Give me one more chance, please.I am</code>
<code>letter.out</code>
<code>I am sorry, I am idiot. Give me one more chance, please.</code>
<code>letter.in</code>
<code>46 33</code> <code>far too longAnd I miss you, been far away for</code>
<code>letter.out</code>
<code>And I miss you, been far away for far too long</code>
<code>letter.in</code>
<code>5 1</code> <code>bcdea</code>
<code>letter.out</code>
<code>abcde</code>

letter.in
5 5 abcde
letter.out
abcde

Обратите внимание, что строка во втором примере входного файла начинается с пробела.

## Задача С. Счастливые билеты strike back!

Имя входного файла: lucky.in  
Имя выходного файла: lucky.out  
Ограничение по времени: 2 секунда  
Ограничение по памяти: 256 мегабайт

Damn, I should be so lucky

---

Jason Mraz — Plane

Пока Шелдон с друзьями был в Петрозаводске, его научили определять, является ли билет в автобусе счастливым. Для этого достаточно проверить, что в номере билета сумма первых трёх цифр равняется сумме трёх последних цифр.

Теперь Шелдон заинтересовался, можно ли из заданных шести цифр составить счастливый билет. Так как у него нет компьютера, он попросил вас помочь ему. Ведущие нули в номерах билетов разрешены.

### Формат входного файла

В единственной строке входного файла через пробел заданы шесть цифр из которых требуется собрать счастливый билет.

### Формат выходного файла

Если из заданных цифр можно составить счастливый билет, то в единственную строку выходного файла выведите номер этого билета(ведущие нули разрешены), иначе выведите -1. Если искомым билетов несколько, то выведите любой из них.

### Примеры

lucky.in	lucky.out
1 2 3 6 0 0	123600
1 2 3 4 5 6	-1
0 0 0 0 0 0	000000

## Задача D. Ох уж эти числа

Имя входного файла: `numbers.in`  
Имя выходного файла: `numbers.out`  
Ограничение по времени: 2 секунда  
Ограничение по памяти: 256 мегабайт

Говард Воловиц увлекается теорией чисел. На одном из форумов, посвященных теории чисел, он нашёл следующую задачу. Дано два целых числа  $n$  и  $m$ , требуется посчитать количество пар чисел  $a$  и  $b$  таких, что  $1 \leq a < b \leq n$  и их сумма кратна  $m$ . Говард легко справился с этой задачей. Справитесь ли вы?

### Формат входного файла

В единственной строке входного файла даны два целых числа  $n$  и  $m$  ( $1 \leq n \leq 10^9, 2 \leq m \leq 10^6$ ).

### Формат выходного файла

В единственной строке выходного файла выведите ответ на задачу.

### Примеры

<code>numbers.in</code>	<code>numbers.out</code>
2 3	1
4 3	2
1 6	0

Примечание:

Обратите внимание, что ответ в этой задаче может быть очень большим, поэтому чтобы избежать переполнения типа, используйте тип данных `long long` в C++ и `int64` в Pascal.

## Задача Е. Сковородки

Имя входного файла: `pans.in`  
Имя выходного файла: `pans.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Большинство людей воспринимают эстетику довольно близко к сердцу: некоторым не нравятся предметы, лежащие не под прямыми углами, некоторым — просто углы, так и Говард Воловиц имеет обычай выравнивать сковородки, расположенные друг на друге, относительно положения их ручек. Небывшая мама часто заставляет его мыть посуду. Иногда, ему приходится мыть такое количество сковородок, что выравнивание вызывает у него головную боль и он вконец запутывается.

Итак, Говард помыл  $n$  сковородок, первая из которых стоит на столе, а каждая  $i$ -ая ( $2 \leq i \leq n$ ) располагается внутри  $(i - 1)$ -ой так, что их центры совпадают. Если посмотреть сверху, можно увидеть круг, из которого в разные стороны торчат ручки. Говард хочет повернуть некоторые сковородки так, чтобы все ручки торчали из стопки в одну сторону.

Обозначим  $\alpha_i$  — угол поворота ручки  $i$ -той сковородки относительно какого-нибудь (одного) направления из центра координат в радианах. Центр координат расположен в центре первой (да и любой другой тоже) сковородки. Говард может повернуть любую сковородку на произвольный угол  $\beta$  в радианах, при этом на тот же угол поворачиваются все сковородки расположенные на той, которую он поворачивает. Ввиду большого количества сковородок, Говард хочет выровнять их за минимальное время. При повороте сковородки на угол  $\beta$  Говард тратит  $\beta$  времени.

Помогите Говарду, составьте оптимальный план поворотов сковородок, требующий минимальные затраты по времени.

### Формат входного файла

В первой строке входного файла задано целое число  $n$  — количество сковородок ( $1 \leq n \leq 10^5$ ). Во второй строке находится  $n$  вещественных чисел  $\alpha_i$  — угол поворота  $\alpha_i \in [0, 2\pi)$  ручки  $i$ -ой сковородки ( $1 \leq i \leq n$ ). Углы поворота заданы не более чем с тремя знаками после запятой.

### Формат выходного файла

Выведите единственное число — минимальное время, требуемое Говарду на преодоление трудности, связанной с неэстетичным расположением сковородок. Ваш ответ будет считаться правильным, если он отличается не более, чем на  $10^{-6}$ , по абсолютной или относительной величине.

### Примеры

<code>pans.in</code>	<code>pans.out</code>
3 0.5 0.234 0.234	0.266
3 0.5 0.2 0.1	0.4

## Задача F. Самолёт

Имя входного файла: `plane.in`  
Имя выходного файла: `plane.out`  
Ограничение по времени: 2 секунда  
Ограничение по памяти: 256 мегабайт

I cannot wait to call you  
And tell you that I landed somewhere  
And hand you a square of the airport

---

Jason Mraz — Plane

Уже всем известные четыре друга Леонард, Шелдон, Говард и Раджеш вылетают из Петрозаводска в Москву. Из петрозаводского аэропорта вылетают самолеты двух типов: Бинг и Бонг. В Бинге кресла располагаются по шесть в один ряд, а в Бонге — по четыре в ряд.

Кресла в Бинге нумеруются следующим образом: номер ряда (число от 10 до 20) и буква от A до F, где A — место у левого окна самолета, C — место у прохода, B — место между A и C, D — место с другой стороны прохода, F — место у правого окна самолета, E — место между D и F.

В Бонге кресла нумеруются похожим образом: номер ряда (число от 1 до 20) и буква от A до D, где A — место у левого окна самолета, B — место у прохода, C — место с другой стороны прохода, D — место у правого окна самолета.

Друзья уже получили свои посадочные талоны с номерами их мест. Теперь они хотят узнать на каком самолете летят.

### Формат входного файла

Во входном файле заданы 4 строки. В  $i$ -ой строке — номер места на  $i$ -ом посадочном талоне в формате, описанном выше.

### Формат выходного файла

В единственной строке выходного файла выведите строку `Bing` если друзья летят на Бинге, `Bong` если они летят на Бонге и `Ambiguous` — иначе.

### Примеры

<code>plane.in</code>	<code>plane.out</code>
1A 1B 1C 1D	Bong
10A 10B 10C 10E	Bing
10A 10B 10C 10D	Ambiguous

## Задача G. Отрезки

Имя входного файла: `segments.in`  
Имя выходного файла: `segments.out`  
Ограничение по времени: 2 секунда  
Ограничение по памяти: 256 мегабайт

Маленький Шелдон на день рождения получил  $n$  отрезков на одной прямой. Каждый отрезок был задан тремя целыми числами  $L, A, B$ . Где  $L$  — какая-то точка на прямой слева от отрезка,  $A$  — расстояние от точки  $L$  до левой границы отрезка,  $B$  — расстояние от точки  $L$  до правой границы отрезка. Шелдону не нравится, что отрезки заданы таким образом. Он хочет, чтобы отрезки были заданы парой целых чисел  $X, D$ . Где  $X$  — точка внутри отрезка,  $D$  — расстояние от точки  $X$  одновременно до левой и до правой границы отрезка.

### Формат входного файла

В первой строке входного файла задано целое число  $n$  ( $1 \leq n \leq 100000$ ) — количество отрезков.

В следующих  $n$  строках записано по три целых числа  $L_i, A_i, B_i$  ( $-10^9 \leq L_i \leq 10^9, 0 \leq A_i \leq B_i \leq 10^9$ ) — описание  $i$ -ого отрезка.

### Формат выходного файла

В выходной файл выведите  $n$  строчек. В  $i$ -ой строке выведите два числа  $X_i, D_i$  — описание  $i$ -ого отрезка, которое нравится Шелдону. Гарантируется, что такое описание всегда будет существовать.

### Примеры

<code>segments.in</code>	<code>segments.out</code>
2	3 1
0 2 4	1 1
-1 1 3	
10	-1 2
-9 6 10	2 0
-5 7 7	13 1
7 5 7	3 0
1 2 2	8 1
0 7 9	-4 0
-9 5 5	-8 1
-10 1 3	8 0
-1 9 9	5 1
-4 8 10	-3 3
-6 0 6	

## Задача Н. Киндер-сюрприз

Имя входного файла: `surprise.in`  
Имя выходного файла: `surprise.out`  
Ограничение по времени: 2 секунда  
Ограничение по памяти: 256 мегабайт

Говард собирает игрушки из киндер-сюрпризов. В последнее время пластиковые контейнеры в киндер-сюрпризах делают из двух половин таким образом, что можно верхнюю половину одного контейнера соединить с нижней половиной второго. Аналогично можно верхнюю половину второго соединить с нижней половиной третьего и так далее.

Говард соединяет  $n$  контейнеров указанным выше образом. При этом последний контейнер он соединяет с первым и получает таким образом кольцо. При взгляде сверху каждый контейнер представляет собой окружность радиуса  $r$ , каждые два соседних контейнера касаются друг друга. При этом центры контейнеров-окружностей сами расположены на одной окружности радиуса  $R$ .

Говарду интересно, какой радиус кольца  $R$  получится, если он соединит  $n$  контейнеров радиуса  $r$ . Помогите ему в этом.

### Формат входного файла

В первой строке входного файла находится два числа — целое число  $n$  ( $3 \leq n \leq 10^6$ ) и вещественное число  $r$  ( $0.01 \leq r \leq 100$ ). Число  $r$  задано не более чем с двумя знаками после десятичной точки.

### Формат выходного файла

Выведите единственное число  $R$  — искомый радиус кольца, которое получится у Говарда. Выводите число с шестью знаками после десятичной точки. Ваш ответ будет признан правильным, если его абсолютная или относительная погрешность будет не более  $10^{-4}$ .

### Примеры

<code>surprise.in</code>	<code>surprise.out</code>
10 1	3.2360679775
3 1.57	1.8128798453

Примечание. Для вывода числа с шестью знаками после десятичной точки, используйте следующие конструкции (предполагается, что выводится переменная  $x$ ).

Pascal:

```
write(x:0:6);
```

C:

```
printf("%.6lf", (double)x);
```

C++:

```
cout.setf(ios::fixed | ios::showpoint);  
cout.precision(6);  
cout << x;
```

## Задача I. Плитка

Имя входного файла: `tiles.in`  
Имя выходного файла: `tiles.out`  
Ограничение по времени: 2 секунда  
Ограничение по памяти: 256 мегабайт

Леонард и Шелдон затеяли ремонт в квартире. Сейчас они ремонтируют ванную комнату и хотят выложить стену кафельной плиткой. Шелдон уже составил «Соглашение о плитке», а Леонард как обычно его не глядя подписал. В Соглашении Шелдон указал следующее:

1. Стена для выкладывания представляет собой прямоугольник размером  $w \times h$  сантиметров.
2. Каждая плитка представляет собой квадрат размера  $a \times a$  сантиметров.
3. Стена должна быть полностью покрыта плиткой. Шириной швов между плитками можно пренебречь.
4. Плитки можно резать только параллельно сторонам. Одну плитку можно разрезать несколько раз.
5. При использовании разрезанных плиток их нужно укладывать так, чтобы линии разреза плиток совпадали с краями стены (другими словами, нельзя допускать того, чтобы линии разреза были внутри стены).

Леонард уже и сам не рад, что подписал такое соглашение. Теперь ему требуется собственно закупить плитку, но он не знает, какое минимальное количество плиток требуется для замощения стены. Помогите ему найти это количество.

### Формат входного файла

Во входном файле находятся три целых числа:  $w, h, a$  — размеры стены и плиток соответственно ( $1 \leq w, h, a \leq 1000$ ).

### Формат выходного файла

Выведите единственное число — минимально возможное количество плиток, которое должен купить Леонард.

### Примеры

<code>tiles.in</code>	<code>tiles.out</code>
2 2 1	4
4 4 3	3
5 5 3	4

На рисунках показано расположение плиток в оптимальном случае во всех примерах. Обратите внимание, что во втором примере для замощения угла приходится использовать отдельную плитку, поскольку разрезанные части плиток должны совпадать с краями стен.

