

Задача 1А. Гость из прошлого

Имя входного файла: **input.txt** или стандартный поток ввода
Имя выходного файла: **output.txt** или стандартный поток вывода
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Коля Герасимов очень любит кефир, и в своём 1984 году он освоил все тонкости покупки этого чудесного напитка. Но однажды, как вам, наверное, известно, он попал в далёкий 2084 год, где покупка кефира представляет собой более сложный процесс.

Будущее будущим, а кушать хочется всегда, поэтому Коля отправился в местную молочную лавку. В 2084 году кефир продают в литровых пластиковых бутылках по a копеек за штуку и в литровых бутылках из стекла по b копеек за штуку. При этом пустую стеклянную бутылку можно сдать и получить назад c ($c < b$) копеек, а пластиковую бутылку сдать нельзя.

У Коли в кармане есть n копеек, и он очень голоден, поэтому хочет выпить как можно больше литров кефира. Так как в его время не было пластиковых бутылок, он совсем не знает, как действовать. Поэтому он обратился за помощью к вам, как к единственному знакомому в будущем.

Формат входных данных

В первой строке входных данных задаётся число n ($1 \leq n \leq 10^{18}$) — количество копеек у Коли в кармане.

В строках со второй по четвертую по одному записаны числа a , b и c ($1 \leq a \leq 10^{18}$, $1 \leq c < b \leq 10^{18}$) — стоимость пластиковой бутылки с кефиром, стоимость стеклянной бутылки с кефиром и сколько копеек можно получить, сдав пустую стеклянную бутылку, соответственно.

Формат выходных данных

Выведите одно целое число — максимальное количество литров кефира, которое сможет выпить Коля.

Примеры

ввод	вывод
10 11 9 8	2
10 5 6 1	2

Пояснение

В первом примере Коля может купить один литр в стеклянной бутылке, затем сдать эту бутылку и снова купить стеклянную бутылку. Таким образом, он сможет выпить два литра кефира.

Во втором примере Коля может купить две пластиковые бутылки и получить два литра кефира или купить сначала один литр в стекле, потом сдать бутылку и купить одну бутылку в пластике. В обоих случаях он купит два литра кефира.

Система оценки

Каждый тест в задаче оценивается отдельно. При этом тесты из условия не оцениваются.

Подзадача 1. Все числа во вводе не превосходят 10 000. Оценивается из 60 баллов.

Подзадача 2. Дополнительные ограничения отсутствуют. Оценивается из 40 баллов. Результаты тестирования данной подзадачи станут доступны после окончания тура.

Задача 1В. Война корпораций

Имя входного файла:	<code>input.txt</code> или стандартный поток ввода
Имя выходного файла:	<code>output.txt</code> или стандартный поток вывода
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Давным-давно в далёкой-далёкой галактике противоборствовали две огромные IT-корпорации: Pineapple и Gogol. Противостояние длится уже многие годы, но близится переломный момент: компания Gogol разработала абсолютно новое устройство, не имеющее аналогов — планшет Lastus 3000.

В этом устройстве используется впервые созданный искусственный интеллект (ИИ). Члены компании Pineapple пытались всеми силами отложить выход нового девайса. В конце концов, за неделю до выхода Lastus 3000 на рынок юристы обнаружили, что название искусственного интеллекта очень похоже на название телефона, который выпустила компания Pineapple 200 лет назад. Так как компания Pineapple обладает авторским правом на это название, она потребовала изменить имя искусственного интеллекта.

Pineapple утверждает, что название их телефона присутствует в качестве подстроки в имени ИИ. Название этой технологии уже было выгравировано на всех планшетах, поэтому директор Gogol предложил вместо некоторых букв в названии ИИ поставить символ «#». Так как это довольно затратно, надо найти минимальное количество символов, которые нужно заменить на «#», чтобы имя ИИ больше не содержало название телефона Pineapple в качестве подстроки. Помогите компании Gogol решить эту задачу.

Подстрокой называется непустая последовательность подряд идущих символов строки.

Формат входных данных

Первая строка входных данных содержит название ИИ компании Gogol, длина названия не превосходит 100 000 символов. Вторая строка входных данных содержит название телефона компании Pineapple, её длина не превосходит 30 символов. Обе строки непустые и содержат только маленькие буквы английского алфавита.

Формат выходных данных

Требуется вывести минимальное количество букв, которое надо заменить на символ «#» в названии ИИ, чтобы оно не содержало название телефона в качестве подстроки.

Примеры

ВВОД	ВЫВОД
intellect tell	1
google apple	0
sirisiri sir	2

Пояснение

В первом примере название ИИ можно заменить на «int#llect».

Во втором примере название можно не менять.

В третьем примере название ИИ можно поменять на «s#ris#ri». Обойтись меньшим количеством замен не получится.

Система оценки

Каждый тест в задаче оценивается отдельно. При этом тесты из условия не оцениваются.

Подзадача 1. Длина первой строки не превосходит 1000. Оценивается из 60 баллов.

Подзадача 2. Дополнительные ограничения отсутствуют. Оценивается из 40 баллов. Результаты тестирования данной подзадачи станут доступны после окончания тура.

Задача 1С. К-специальные таблички

Имя входного файла: **input.txt** или стандартный поток ввода
Имя выходного файла: **output.txt** или стандартный поток вывода
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Чего не сделаешь ради того, чтобы выделиться из серой массы. Кто-то танцует, кто-то учит наизусть правила русского языка, кто-то пытается стать выдающимся спортивным программистом, ну а кто-то коллекционирует забавные математические объекты.

Алиса как раз из таких коллекционеров. Сейчас она очень хочет заполучить k -специальную табличку. Напомним, что табличка $n \times n$ называется k -специальной, если выполнены следующие три условия:

- каждое число от 1 до n^2 встречается в ней ровно 1 раз;
- в каждой строке числа идут в возрастающем порядке;
- сумма чисел, расположенных в k -м столбце, максимальна.

Помогите Алисе! Отыщите хотя бы одну k -специальную табличку размера $n \times n$. Строки и столбцы нумеруются от 1 до n , при этом строки нумеруются сверху вниз, а столбцы слева направо.

Формат входных данных

В первой строке входных данных записаны два числа n и k ($1 \leq n \leq 500, 1 \leq k \leq n$) — размер искомой таблички и номер столбца, сумма в котором должна быть максимальна.

Формат выходных данных

В первой строке выведите сумму чисел в k -м столбце искомой таблички.

В следующих n строках выведите саму табличку: сначала n элементов первой строки, затем n элементов второй и так далее.

Если искомым табличек несколько, то выведите любую из них.

Примеры

ввод	вывод
4 1	28 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
5 3	85 5 6 17 18 19 9 10 23 24 25 7 8 20 21 22 3 4 14 15 16 1 2 11 12 13

Система оценки

Каждый тест в задаче оценивается отдельно. При этом тесты из условия не оцениваются.

Подзадача 1. Во всех тестах данной подзадачи n не превосходит 6. Оценивается из 38 баллов.

Подзадача 2. Дополнительные ограничения отсутствуют. Оценивается из 62 баллов. Результаты тестирования данной подзадачи станут доступны после окончания тура.

Задача 2. Контрольная по арифметике

Имя входного файла:	<code>input.txt</code> или стандартный поток ввода
Имя выходного файла:	<code>output.txt</code> или стандартный поток вывода
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Отличник Витя учится в третьем классе. На последнем уроке математики в четверти учительница Оксана Филипповна дала итоговую контрольную работу по арифметике. На контрольной сообразительный Витя быстрее всех справился со стандартными заданиями. Поскольку до конца урока оставалось ещё много времени, учительница предложила ему решить более сложное упражнение.

Рассмотрим *операцию переворота* целого положительного числа: при её применении десятичные цифры числа записываются в обратном порядке, затем если у результата есть ведущие нули, они отбрасываются. Например, при перевороте из числа 123 получается число 321, из числа 130 — число 31, а из числа 31 — число 13.

Оксана Филипповна загадала целое положительное число a без ведущих нулей и применила к нему операцию переворота, получилось число a_r . После этого она сложила a и a_r , дала Вите их сумму n и попросила его найти исходное число a .

Поскольку в предложенном учительницей примере числа a и a_r были небольшими, Витя быстро угадал правильный ответ и задумался над построением общего алгоритма для решения задачи. Его интересует, как по заданному числу n найти какое-либо a или определить, что такого a не существует. Помогите Вите разработать такой алгоритм.

Формат входных данных

В первой строке входных данных находится одно целое число n ($1 \leq n \leq 10^{100\,000}$).

Формат выходных данных

В случае если не существует ни одного целого положительного числа a без ведущих нулей такого, что $a + a_r = n$, выведите единственное число 0.

В противном случае выведите подходящее a без пробелов и ведущих нулей. Если удовлетворяющих условию вариантов несколько, выведите любой.

Для лучшего понимания формата вывода рекомендуется изучить примеры.

Примеры

ВВОД	ВЫВОД
4	2
11	10
5	0
33	21

Пояснение

В первом примере $4 = 2 + 2$, $a = 2$ — единственный вариант.

Во втором примере $11 = 10 + 1$, $a = 10$ — единственный вариант. Обратите внимание, что вариант $a = 01$ не подходит, поскольку a не может иметь ведущие нули.

Легко проверить, что в третьем примере ни одного подходящего a не существует.

В четвёртом примере $33 = 30 + 3 = 12 + 21$, и существует три варианта для a : $a = 30$, $a = 12$, $a = 21$. Любой из них является правильным ответом.

Система оценки

Тесты к этой задаче состоят из пяти групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов **предыдущих** групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Тесты	Баллы	Дополнительные ограничения	Комментарий
			n	
0	1 – 4	0	–	Тесты из условия.
1	5 – 36	20	$n \leq 1\,000\,000$	
2	37 – 59	20	$n \leq 10^{10}$	
3	60 – 81	20	$n \leq 10^{25}$	
4	82 – 102	40	$n \leq 10^{100\,000}$	Offline-проверка.

Задача 3. лягушачьи бои

Имя входного файла:	<code>input.txt</code> или стандартный поток ввода
Имя выходного файла:	<code>output.txt</code> или стандартный поток вывода
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Недавно великий комбинатор посетил лягушачьи бои и, поразившись несистемностью правил боёв, решил создать своё увлекательное зрелище, но подчинённое строгим правилам.

Итак, лягушек выставляют на поле, имеющее форму кольца, разделённого на m клеток. Клетки пронумерованы от 1 до m по кругу, после клетки номер m идёт клетка номер 1. i -я лягушка изначально может прыгнуть на a_i клеток вперёд.

лягушки ходят по очереди, начиная с лягушки с номером 1. На своём ходу i -я лягушка прыгает на a_i клеток вперёд, сбрасывая с поля всех других лягушек на своём пути, в том числе и ту, которая находится в конечной клетке текущего прыжка. После этого её значение a_i уменьшается на количество удалённых во время данного прыжка лягушек. Если a_i становится равным нулю или отрицательным, то i -я лягушка не сможет прыгать в дальнейшем.

После того, как лягушка с номером 1 заканчивает свой ход, ходит лягушка с номером 2, затем лягушка с номером 3 и так далее. После того, как сходит лягушка с номером n , снова ходит лягушка с номером 1. Если лягушка с каким-то номером уже была сброшена с поля, то считается, что она всегда пропускает свой ход.

Великому комбинатору уже не терпится узнать, какие из лягушек в конце концов останутся на поле. Помогите ему узнать это.

Формат входных данных

В первой строке содержатся два числа n и m ($1 \leq n \leq 100\,000$, $2 \leq m \leq 10^9$, $n \leq m$) — количество лягушек и размер поля соответственно.

В следующих n строках содержатся описания лягушек. Каждое описание состоит из двух чисел p_i и a_i ($1 \leq p_i, a_i \leq m$) — номер клетки, на которой стоит лягушка, и начальная длина её прыжка соответственно.

Гарантируется, что все значения p_i различны.

Формат выходных данных

В первой строке выведите количество лягушек, которые всегда будут на поле.

В следующей строке выведите номера этих лягушек в любом порядке.

Примеры

ВВОД	ВЫВОД
3 5 2 1 5 3 4 3	1 3
5 6 1 2 3 4 2 5 5 1 6 1	2 1 4

Пояснение

В первом примере первая лягушка прыгает на одну клетку и оказывается в клетке номер 3. Затем вторая прыгает на 3 клетки и также оказывается в клетке номер 3, удаляя при этом первую лягушку с поля. Величина следующего прыжка у неё теперь равна 2. Третья лягушка прыгает в клетку номер 2. Затем вторая лягушка прыгает в клетку 5. Третья догоняет её и удаляет с поля. На поле осталась только она.

Во втором примере первая лягушка прыгает на две клетки, удаляя с поля лягушек, стоящих на клетках 2 и 3. Величина её следующего прыжка при этом становится равной нулю. Затем прыгает четвёртая лягушка, удаляет с поля пятую и тоже останавливается. В итоге на поле остаются две лягушки.

Система оценки

Каждый тест в данной задаче оценивается отдельно. При этом тесты из условия не оцениваются.

Подзадача 1. Во всех тестах данной подзадачи $n, m \leq 10$. Оценивается из 20 баллов.

Подзадача 2. Во всех тестах данной подзадачи $n, m \leq 100$. Оценивается из 15 баллов.

Подзадача 3. Во всех тестах данной подзадачи $n, m \leq 100\,000$. Оценивается из 15 баллов.

Подзадача 4. Во всех тестах данной подзадачи $n \leq 1\,000$. Оценивается из 30 баллов. Результаты тестирования данной подзадачи станут доступны после окончания тура.

Подзадача 5. Дополнительные ограничения отсутствуют. Данная подзадача оценивается из 20 баллов. Результаты тестирования данной подзадачи станут доступны после окончания тура.