

Задача А. Банковские счета

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

У банка есть клиенты. Каждый клиент имеет ровно один счет. Периодически балансы счетов клиентов изменяются, а также иногда клиенты хотят узнать, сколько денег у них на счету.

Напишите программу, которая будет обрабатывать запросы двух видов:

1. Баланс клиента s_i изменяется на величину x_i (увеличивается, если $x_i > 0$, и уменьшается, если $x_i < 0$). Если до этого момента баланс клиента s_i ни разу не изменялся, его баланс становится равным x_i .
2. Клиент s_i хочет узнать, какая сумма в данный момент есть на его счету. Если до этого момента баланс клиента s_i ни разу не изменялся, следует сообщить об ошибке.

Суммы на балансах клиентов в любой момент могут быть как положительными, так и отрицательными.

Формат входных данных

В первой строке входных данных содержится число n — количество запросов ($1 < n < 100\,000$). Далее следуют n строк, каждая из которых содержит описание запроса одного из двух типов.

Первое число в строке описывает тип запроса. Если это число 1, то далее в строке следует строка s_i и целое число x_i — имя клиента и изменение баланса соответственно ($|x_i| \leq 10\,000$). Если первое число равно 2, то далее в строке следует строка s_i — имя клиента, который хочет узнать свой баланс.

Формат выходных данных

Для каждого запроса 2-го типа выведите в отдельной строке текущий баланс для заданного клиента. Если на момент запроса не было ни одного изменения баланса этого клиента, выведите вместо этого строку «ERROR».

Пример

стандартный ввод	стандартный вывод
7	3
1 asdf 3	1
1 zxcv 5	ERROR
2 asdf	5
1 asdf -2	
2 asdf	
2 lalala	
2 zxcv	

Задача В. Тараканы

Имя входного файла: `cockroach.in`
Имя выходного файла: `cockroach.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

64 мегабайта

Пытаясь залезть в холодильник, таракан Серёга не успел увернуться от разящего тапка пенсионерки Агафьи Петровны и погиб геройской смертью. Поэтому в ближайшую тёмную ночь оставшиеся тараканы собрались на кухне, чтобы отомстить и устроить крестовый поход на холодильник. Они построились в колонну по два таракана в ряду. Но в последнем ряду осталось место для Серёги. Тогда они перестроились в колонну по три таракана, но опять одного не хватило. Каждый раз они перестраивались так, чтоб на одного таракана в ряду было больше, чем в прошлый раз, и каждый раз им не хватало Серёги.

Какое минимальное число тараканов может проживать в квартире у Агафьи Петровны, если до того момента, как она встала и пришла на кухню, чтобы приготовить себе завтрак, тараканы успели перестроиться N раз?

Формат входных данных

Дано натуральное число N ($N \leq 10\,000$).

Формат выходных данных

Требуется вывести одно число — ответ на вопрос.

Пример

<code>cockroach.in</code>	<code>cockroach.out</code>
4	59

Замечание

Пояснение к примеру 1.

- 1 перестроение. По два таракана в ряду.
- 2 перестроение. По три таракана в ряду.
- 3 перестроение. По четыре таракана в ряду.
- 4 перестроение. По пять тараканов в ряду.

Задача С. Связность графа

Имя входного файла: `connected.in`
Имя выходного файла: `connected.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

64 мегабайта

Дан неориентированный граф, содержащий N вершин и M рёбер ($1 \leq N \leq 1000$, $0 \leq M \leq 7000$). Требуется найти наименьшее число рёбер и эти рёбра, которые нужно добавить, чтобы граф стал связным.

Формат входных данных

Во входном файле записаны сначала числа N и M , затем идёт описание рёбер графа — M пар чисел, где каждая пара описывает начало и конец ребра.

Формат выходных данных

В первую строку вывести единственное число K — минимальное число рёбер, которое нужно добавить. В следующих K строках выведите по 2 числа — начало и конец нового ребра. Разрешается выводить любые рёбра, удовлетворяющие условию.

Пример

<code>connected.in</code>	<code>connected.out</code>
3 1 2 1	1 1 3

Задача D. Эклеры

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

После успешной сдачи всех зачетов Вася купил себе в подарок коробку, содержащую n сладких эклеров. Вася решил каждое утро есть некоторое одинаковое число эклеров, пока они все не закончатся. Однако сосед Васи, Петя, заметил принесенную Васей коробку и тоже решил насладиться вкусом эклеров.

Теперь процесс поедания эклеров выглядит следующим образом: сначала Вася выбирает число k , одинаковое для всех дней. Затем утром он съедает k эклеров из коробки (или доедает все эклеры, если их осталось меньше k), после этого Петя вечером съедает 10% **оставшихся** эклеров. Если эклеры еще не закончились, то на следующий день Вася опять съедает k эклеров, а Петя — 10% от оставшихся и так далее.

Если число эклеров не делится на 10, то Петя округляет «свою» долю в меньшую сторону, например, если в коробке было 97 эклеров, то Петя съест только 9 из них. В частности, если в коробке уже меньше 10 эклеров, то Петя не будет их есть вообще.

Определите, какое наименьшее число k может выбрать Вася такое, что он съест не менее половины всех n эклеров, которые были в коробке изначально. Заметьте, что число k должно быть натуральным.

Формат входных данных

В первой строке содержится натуральное число n ($1 \leq n \leq 10^{18}$) — изначальное количество эклеров.

Формат выходных данных

Вывести единственное число — наименьшее значение k , удовлетворяющее Васю.

Примеры

стандартный ввод	стандартный вывод
68	3
1	1

Задача Е. Делители

Имя входного файла: `onedivisor.in`
Имя выходного файла: `onedivisor.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Вася любит простые числа, а особенно он любит простые числа, на которые делится его самое любимое число K .

Формат входных данных

На вход подается единственное целое положительное число K отличное от 1, не превосходящее $3 * 10^9$.

Формат выходных данных

В первой строке выведите количество чисел, которые Вася особенно любит. Во второй строке выведите в порядке возрастания все простые числа, которые Вася особенно любит.

Примеры

<code>onedivisor.in</code>	<code>onedivisor.out</code>
5	1 5
12	2 2 3
30	3 2 3 5

Задача F. Выходные дни

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

В некотором государстве год состоит из N дней, причём неделя состоит из w подряд идущих рабочих дней. Профсоюзы требуют, чтобы после w рабочих дней было добавлено h выходных дней, тем самым неделя будет состоять из $w + h$ дней — сначала w рабочих дней, затем h выходных, снова w рабочих дней, затем h выходных и т.д. Последняя неделя может быть неполной (если N не делится на $w + h$), тогда в неполной неделе сначала идут рабочие дни (не более w), затем выходные.

Олигархи согласны на такую реформу календаря, но настаивают, чтобы общее число рабочих дней в году было не менее M . Определите, какое наибольшее число выходных дней в каждой неделе можно добавить в календарь так, чтобы выполнить требования олигархов.

Формат входных данных

Программа получает на вход три целых числа N , M , w — записанных в одной строке через пробел. N — количество дней в году, M — минимальное общее число рабочих дней в году, w — число рабочих дней в одной неделе. $1 \leq N \leq 10^{18}$, $1 \leq M \leq N$, $1 \leq w < M$.

Формат выходных данных

Программа должна вывести одно целое число — максимальное число выходных дней, которое можно добавить в неделю так, что общее число рабочих дней в году будет не менее M .

Примеры

<code>holidays.in</code>	<code>holidays.out</code>
100 70 8	3
10 3 2	7

Задача G. Беги, Альф! Беги!

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Прямо сейчас Альфу снится кошмар. В нем он бежит по дороге с препятствиями, на которой, ко всему прочему, разбросаны монеты.

Дорога представляет из себя таблицу $n \times 3$, в клетках которой либо ничего нет, либо находится стена, либо монета. Альф бежит вдоль стороны длиной n . Начинает он бежать из первой строки (то есть у него есть три варианта начала, он может выбрать любой из них) и бежит до тех пор, пока не врежется в стену, либо не пробежит дорогу целиком (не окажется в строчке n).

Пусть сейчас Альф стоит в строке x и столбце y — $(x; y)$, тогда он может попасть в три возможные клетки: $(x + 1; y - 1)$, $(x + 1; y)$, $(x + 1; y + 1)$, если конечно новая клетка не выходит за пределы дороги, и в ней не находится стена. Так как все обитатели планеты Мелмак умеют контролировать свои сны, Альф смог получить карту дороги. Теперь он хочет узнать, какое наибольшее количество монет можно собрать к концу забега.

Так как контроль сна отнимает у Альфа много сил, он просит вас написать программу, которая по карте сможет определить наибольшее количество монет, которое можно собрать за один забег.

Формат входных данных

В первой строке входного файла задано число n ($1 \leq n \leq 10^4$) — количество строк в таблице. В следующих n строках дано по три символа, характеризующие данную строку таблицы. $.$ равен «.», если клетка пустая, C , если в этой клетке монета, и W , если стена. Если в первой строке во всех клетках находятся стены, Альф заканчивает забег сразу.

Формат выходных данных

В выходной файл выведите одно число — наибольшее количество монеток, которые можно собрать.

Примеры

стандартный ввод	стандартный вывод
5 W.W C.C WW. CC. CWW	3
4 W.W CWC W.W CWW	2

Задача Н. Кинотеатр

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Если у вас так много времени, что вы не знаете, чем заняться, все друзья куда-то уехали, сидеть дома не хочется, то можно попробовать сходить в кинотеатр. Маша уже там. И через 15 минут начнётся показ фильма «Крестики-нолики: жажда возмездия».

Но сперва надо купить билет. В кинотеатре «Кинорандом», куда пришла Маша, ровно один зал, в котором ровно один ряд с местами, пронумерованными от 1 до S . В кассу стоит очередь. Маша посчитала, что её порядковый номер в очереди — это U . То есть перед ней стоит $U - 1$ человек. Маша знает, что каждый посетитель в очереди (включая её саму) покупает один билет и при выборе места придерживается следующего правила: он садится как можно ближе к центру ряда, и если таких мест несколько, то выбирается место с наименьшим номером.

Маша знает, какие места были заняты в тот момент, когда она становилась в очередь. Теперь её интересует номер места, на котором она будет смотреть кино.

Формат входных данных

В первой строке записано положительное целое число U ($1 \leq U \leq 100\,000$) — количество людей в очереди, включая Машу. Во второй строке через пробел указаны целые числа S ($1 \leq S \leq 10^9$) и G ($0 \leq G \leq 100\,000$). Здесь S — это общее количество мест в ряду, а G — количество групп, которые описывают подряд идущие занятые места в тот момент, когда Маша становилась в очередь. Следующие G строк содержат по два числа a_i, b_i ($1 \leq a_i \leq b_i \leq S$). Каждая пара чисел a_i, b_i определяет, что места с a_i по b_i включительно заняты.

Гарантируется, что группы занятых мест не пересекаются и описываются во входных данных в возрастающем порядке своих начал. Гарантируется, что в ряду есть как минимум U свободных мест.

Формат выходных данных

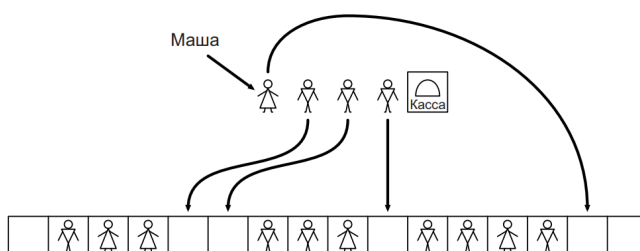
Выведите номер места, на которое сядет Маша.

Примеры

стандартный ввод	стандартный вывод
1 30 0	15
4 16 4 2 3 4 4 7 9 11 14	15

Замечание

Иллюстрация ко второму тесту:



Задача I. Спутник

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

Компания «РосПрог» занимается написанием программного обеспечения для спутников. Спутники летают быстро, поэтому и программы на нём должны работать быстро (иначе он может не успеть, например, рассчитать и подкорректировать свою траекторию). У разработчиков есть несколько различных реализаций функции расчёта траектории, и они хотят выбрать самую быструю из них.

Для этого они взяли k реализаций, запустили каждую по n раз на тестовом стенде и измерили, сколько времени эти реализации каждый раз работали. После этого для каждой пары реализаций a и b было посчитано доминирование a над b . Доминированием реализации a над реализацией b называется количество пар запусков реализаций a и b таких, что запуск реализации a отработал строго медленнее запуска реализации b .

После этого была посчитана производительность каждой реализации. Производительность реализации a определяется как сумма доминирований a над всеми реализациями, кроме a . Из посчитанных данных должен быть составлен отчёт для начальства, но в последний день перед сдачей данные были потеряны. Помогите разработчикам всё-таки сдать отчёт начальству и восстановите значения всех производительностей.

Формат входных данных

В первой строке задано два числа n и k ($1 \leq n, k \leq 1000$) — количество запусков и количество различных реализаций, соответственно. Далее, в k строках задано по n целых чисел $a_{i,j}$ ($1 \leq a_{i,j} \leq 10^9$) — время работы j -го запуска i -й реализации.

Все $a_{i,j}$ различны.

Формат выходных данных

В первой и единственной строке выведите k чисел. i -е число должно равняться производительности i -й реализации.

Примеры

стандартный ввод	стандартный вывод
3 3 1 4 7 2 5 8 3 6 9	6 9 12
1 3 2 3 9	0 1 2

Задача J. Зелья

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В игре «Информляндия. Победи монстра - 2» используются зелья силы $z_1, z_2, \dots, z_N$. Школьник пришел на бой к финальному боссу и обнаружил, что у него есть ровно по две склянки каждого зелья. Чтобы победить монстра надо принять зелий на сумму ровно S . Напишите программу, определяющую, сможет ли школьник сдать экзамен.

Формат входных данных

Сначала вводится число S , ($1 \leq S \leq 10^9$), затем — число N ($1 \leq N \leq 15$) и далее N попарно различных чисел $z_1, z_2, \dots, z_N$ ($1 \leq z_i \leq 10^9$).

Формат выходных данных

Если у школьника не хватит зелий, чтобы победить монстра, выведите одно число -1 (минус один). Если нельзя набрать нужную силу полными склянками, то выведите 0. Иначе выведите 1.

Примеры

стандартный ввод	стандартный вывод
5 2 1 2	1
7 2 1 2	-1
5 2 3 4	0

Задача К. Доить нельзя не доить!

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Каждый день Фермер Джон доит своих 8 коров, которых зовут Bessie, Buttercup, Belinda, Beatrice, Bella, Blue, Betsy, и Sue. К несчастью, коровы довольно разборчивы и требуют, чтобы ФД доил их в порядке, который соответствует N ограничениям ($1 \leq N \leq 7$). Каждое из ограничений имеет вид «X must be milked beside Y» («X необходимо подоить рядом Y»), что означает, что в порядке дойки корову X нужно доить сразу после коровы Y или непосредственно перед коровой Y .

Пожалуйста, помогите ФД определить порядок дойки его коров, который удовлетворяет всем требуемым ограничениям. Гарантируется, что такое упорядочивание всегда возможно. Если возможно несколько упорядочиваний, выведите первое из них в алфавитном порядке. То есть, первая корова должна иметь наименьшее в алфавитном порядке имя из всех возможных имён, которые могут быть первыми. Среди всех упорядочиваний, начинающихся с этой же первой коровы, вторая корова должна быть наименьшей в алфавитном порядке среди всех корректных упорядочиваний и т.д.

Формат входных данных

Первая строка ввода содержит N . Каждая из последующих N строк содержит предложение описывающее ограничение вида «X must be milked beside Y» где X и Y - имена некоторых их коров ФД (8 возможных вариантов перечислены в начале условия задачи).

Формат выходных данных

Выведите, используя 8 строк, порядок имён коров, по одному в строке, удовлетворяющий всем ограничениям. Если возможно несколько порядков, выведите наиболее ранний в смысле алфавита.

Пример

стандартный ввод	стандартный вывод
3 Buttercup must be milked beside Bella Blue must be milked beside Bella Sue must be milked beside Beatrice	Beatrice Sue Belinda Bessie Betsy Blue Bella Buttercup