

Задача А. Словарь

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

Кхамул решил составить толковый словарь орочей речи. Он последовательно приставал к N оркам, каждый говорил ему слово по-орочьи, и Кхамул записывал это слово. Некоторые из записанных слов совпали. Определений к словам Кхамул решил не писать: все равно каждый орк знает их значение. Таким образом, все, что вам осталось (Кхамулу эта затея уже надоела) — отсортировать список слов в лексикографическом порядке и, при наличии одинаковых слов, оставить из них только одно.

Формат входных данных

В первой строке находится целое число N — число слов ($1 \leq N \leq 100$). В следующих N строках находятся слова орочьего языка, состоящие только из больших букв. Длины слов не превышают 100.

Формат выходных данных

Выведите готовый словарь Кхамула — N неповторяющихся орочьих слов в лексикографическом порядке.

Пример

dictionary.in	dictionary.out
Б	А
С	В
С	С
А	
В	
А	

Задача В. 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 С. 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 D. Телефонные номера

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

Однажды Леонид захотел позвонить Владилену, но вдруг обнаружил, что не помнит его номера телефона. Всё, что ему удалось вспомнить про номер Владилены, — это то, что он состоит ровно из N цифр, сумма которых равняется K .

Теперь Леонид хочет узнать, сколько существует телефонных номеров длины N с суммой цифр K , состоящих из цифр от 0 до 9.

Формат входных данных

В первой строке входного файла записаны два числа N и K ($1 \leq N \leq 200$, $0 \leq K \leq 9N$).

Формат выходных данных

Выведите количество телефонных номеров, удовлетворяющих данным требованиям.

Пример

<code>tele.in</code>	<code>tele.out</code>
3 2	6

Задача Е. 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 F. 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 G. 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 Н. 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 I. 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