

## Задача А. Коровы - в стойла

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

На прямой расположены стойла, в которые необходимо расставить коров так, чтобы минимальное расстояние между коровами было как можно больше.

### Формат входных данных

В первой строке вводятся числа  $N$  ( $2 < N \leq 10^5$ ) – количество стойл и  $K$  ( $1 < K < N$ ) – количество коров. Во второй строке задаются  $N$  натуральных чисел в порядке возрастания – координаты стойл (координаты не превосходят  $10^9$ ).

### Формат выходных данных

Выведите одно число – наибольшее возможное допустимое расстояние.

### Примеры

стандартный ввод	стандартный вывод
6 3 2 5 7 11 15 20	9
5 3 1 2 3 100 1000	99

## Задача В. Квадратный корень и квадратный квадрат

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

Найдите такое число  $x$ , что  $x^2 + \sqrt{x} = C$ , с точностью не менее 6 знаков после точки.

### Формат входных данных

В единственной строке содержится вещественное число  $1 \leq C \leq 10^{10}$ .

### Формат выходных данных

Выведите одно число — искомый  $x$ .

### Примеры

стандартный ввод	стандартный вывод
2.0000000000	1.00000000000000000000
18.0000000000	4.00000000000000000000

## Задача С. When democracy fails

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

В одной демократической стране приближаются парламентские выборы. Выборы проходят по следующей схеме: каждый житель страны, достигший восемнадцатилетнего возраста, отдает свой голос за одну из политических партий. После этого партия, которая набрала максимальное количество голосов, считается победившей на выборах и формирует правительство. Если несколько партий набрали одинаковое максимальное количество голосов, то они должны сформировать коалиционное правительство, что обычно приводит к длительным переговорам.

Один бизнесмен решил выгодно вложить свои средства и собрался поддержать на выборах некоторые партии. В результате поддержки он планирует добиться победы одной из этих партий, которая затем сформирует правительство, которое будет действовать в его интересах. При этом возможность формирования коалиционного правительства его не устраивает, поэтому он планирует добиться строгой победы одной из партий.

Чтобы повлиять на исход выборов, бизнесмен собирается выделить деньги на агитационную работу среди жителей страны. Исследование рынка показало, что для того, чтобы один житель сменил свои политические воззрения, требуется потратить одну условную единицу. Кроме того, чтобы  $i$ -я партия в случае победы сформировала правительство, которое будет действовать в интересах бизнесмена, необходимо дать лидеру этой партии взятку в размере  $p_i$  условных единиц. При этом некоторые партии оказались идеологически устойчивыми и не согласны на сотрудничество с бизнесменом ни за какие деньги.

По результатам последних опросов известно, сколько граждан планируют проголосовать за каждую партию перед началом агитационной компании. Помогите бизнесмену выбрать, какую партию следует подкупить, и какое количество граждан придется убедить сменить свои политические воззрения, чтобы выбранная партия победила, учитывая, что бизнесмен хочет потратить на всю операцию минимальное количество денег.

### Формат входных данных

В первой строке вводится целое число  $n$  – количество партий ( $1 \leq n \leq 10^5$ ). Следующие  $n$  строк описывают партии. Каждая из этих строк содержит по два целых числа:  $r_i$  – количество жителей, которые собираются проголосовать за эту партию перед началом агитационной компании, и  $b_i$  – взятка, которую необходимо дать лидеру партии для того, чтобы сформированное ей в случае победы правительство действовало в интересах бизнесмена ( $1 \leq r_i \leq 10^6$ ,  $1 \leq b_i \leq 10^6$  или  $b_i = -1$ ). Если партия является идеологически устойчивой, то  $b_i = -1$ . Гарантируется, что хотя бы одно  $b_i$  не равно  $-1$ .

### Формат выходных данных

В первой строке выведите минимальную сумму, которую придется потратить бизнесмену. Во второй строке выведите номер партии, лидеру которой следует дать взятку. В третьей строке выведите  $n$  целых чисел – количество голосов, которые будут отданы за каждую из партий после осуществления операции. Если оптимальных решений несколько, выведите любое.

### Примеры

стандартный ввод	стандартный вывод
3 7 -1 2 8 1 2	6 3 3 2 5
2 239 239 238 -1	239 1 239 238

## Задача D. Последний рубеж

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

### Это интерактивная задача.

Однажды общительный программист Павел позвал своих друзей на квест. Ребята с лёгкостью решали головоломки и продвигались вперёд. И вот им осталось решить последнюю загадку перед тем, как получить долгожданный приз.

Загадка состоит в том, что перед ребятами находится дверь с  $N$  замками, которую нужно открыть. Некоторые замки открыты, а некоторые закрыты. Ребята не знают, какие замки уже открыты, однако, потратив некоторое время на изучение одного конкретного замка, они могут определить, открыт он или нет. Рядом с дверью висит табличка, на которой написано, что самый левый замок открыт, а самый правый закрыт.

В процессе выполнения предыдущих заданий ребята выяснили, как открыть дверь. Пронумеруем замки слева направо числами от 1 до  $N$ . Тогда для того, чтобы открыть дверь, ребятам нужно найти замок с номером  $i < N$ , такой что замок  $i$  открыт, а замок  $i + 1$  закрыт.

Как уже было сказано, для того, чтобы определить, является ли  $i$ -й замок открытым, им нужно подробно его осмотреть, потратив на это некоторое время. Так как у ребят осталось немного времени для выполнения последнего задания, они могут подробно осмотреть не более  $Q$  замков.

Помогите ребятам открыть дверь.

### Протокол взаимодействия

В начале на вход программе подаётся одно целое число  $N$  ( $2 \leq N \leq 10^{18}$ ).

После этого вы можете делать запросы вида `? i`, означающие, что ребята подробно осматривают замок с номером  $i$ . В ответ на подобный запрос вы получите число 0, означающее, что замок закрыт, или число 1 в противном случае.

Если вы нашли ответ, вы должны сделать запрос вида `! i`, означающий, что вы считаете, что замок  $i$  открыт, а замок  $i + 1$  закрыт. После этого запроса вы должны завершить работу программы.

Считается, что в начале вы знаете состояние замков 1 и  $N$ .

Вы должны сделать не более  $Q$  запросов первого типа. В случае превышения этого ограничения ваше решение получит вердикт «Неправильный ответ».

В случае нарушения каких-либо правил взаимодействия с программой-интерактором, ваше решение может получить любой вердикт.

После каждого запроса, в том числе после запроса второго типа, вы должны выполнить операцию `flush`.

Для сброса буфера вывода (то есть для операции `'flush'`) сразу после вывода запроса и перевода строки нужно сделать:

- `fflush(stdout)` или `cout.flush()` в языке C++;
- `System.out.flush()` в Java;
- `stdout.flush()` в Python;
- `flush(output)` в Pascal;
- смотрите документацию для других языков.

Если вы не сделаете операцию `flush` после какого-либо запроса, ваше решение может получить любой вердикт.

## Система оценки

Подзадача	Баллы	Доп. ограничения	Оценка	Необх. подзадачи
0	0	Тесты из условия	потестовая	—
1	20	$Q = 100 \ 2 \leq N \leq 100$	подзадача	—
2	10	$Q = 100 \ 2 \leq N \leq 10^5$ В начале закрыты ровно два замка	подзадача	—
3	30	$Q = 90 \ 2 \leq N \leq 10^9$	подзадача	1, 2
4	20	$Q = 65 \ 2 \leq N \leq 10^{18}$	подзадача	1, 2, 3
5	20	$Q = 60 \ 2 \leq N \leq 10^{18}$	подзадача	1, 2, 3, 4

Для каждой подзадачи сообщаются набранные баллы, а также результат тестирования на первом непройденном тесте.

## Примеры

стандартный ввод	стандартный вывод
3 0	? 2 ! 1
3 1	? 2 ! 2
5 0 0 1	? 2 ? ? 3 ? ? 4 ! ! 4

## Задача Е. Минимизируем максимум

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 4 секунды  
Ограничение по памяти: 512 мегабайт

Даны  $n$  нестрого возрастающих массивов  $A_i$  и  $m$  нестрого убывающих массивов  $B_j$ . Все массивы имеют одну и ту же длину  $l$ . Далее даны  $q$  запросов вида  $(i, j)$ , ответ на запрос – такое  $k$ , что  $\max(A_{ik}, B_{jk})$  минимален. Если таких  $k$  несколько, можно вернуть любое.

### Формат входных данных

На первой строке числа  $n, m, l$  ( $1 \leq n, m \leq 900; 1 \leq l \leq 3000$ ). Следующие  $n$  строк содержат описания массивов  $A_i$ . Каждый массив описывается перечислением  $l$  элементов. Элементы массива – целые числа от 0 до  $10^5 - 1$ . На следующих  $m$  строках идёт описание массивов  $B_j$  в таком же формате. На следующей строке число запросов  $q$  ( $1 \leq q \leq n \cdot m$ ). Следующие  $q$  строк содержат пары чисел  $i, j$  ( $1 \leq i \leq n, 1 \leq j \leq m$ ).

Массивы и элементы внутри массива нумеруются с 1.

### Формат выходных данных

Выведите  $q$  чисел от 1 до  $l$  – ответы на запросы.

### Пример

стандартный ввод	стандартный вывод
4 3 5	3
1 2 3 4 5	4
1 1 1 1 1	3
0 99999 99999 99999 99999	5
0 0 0 0 99999	4
5 4 3 2 1	3
99999 99999 99999 0 0	1
99999 99999 0 0 0	2
12	2
1 1	4
1 2	4
1 3	3
2 1	
2 2	
2 3	
3 1	
3 2	
3 3	
4 1	
4 2	
4 3	

## Задача F. Поиск позиции

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

В шеренгу друг за другом стоят  $n$  человек, рост  $i$ -го из них равен  $a_i$  условных единиц. Вы тоже собираетесь встать в эту шеренгу, при чем вам хочется встать на такую позицию  $p$ , чтобы  $f(p) =$  [количество людей слева от вас того же роста, что и вы] умножить на [количество людей справа от вас с ростом, не равным вашему] было максимально.

Для этого вы можете встать в начало шеренги, в её конец, или между любыми двумя соседними людьми.

К сожалению вы не можете точно вспомнить ваш рост, у вас есть только  $m$  предположений о том, каким он может быть, и для каждого из них вы хотели бы знать оптимальную позицию, на которую вам стоило бы встать.

### Формат входных данных

В первой строке даны два целых числа  $n$  и  $m$  ( $1 \leq m, n \leq 10^5$ ). Во второй строке даны  $n$  целых чисел  $a_i$  - рост людей, стоящих в шеренге ( $1 \leq a_i \leq 10^5$ ) В третьей строке даны  $m$  целых чисел  $x_i$  - ваш предполагаемый рост ( $1 \leq x_i \leq 10^5$ )

### Формат выходных данных

В единственной строке выведите выведите  $m$  целых чисел - значение  $f(p)$  в оптимальной для данного роста позиции.

### Примеры

стандартный ввод	стандартный вывод
4 3 1 1 2 3 1 2 4	4 1 0
9 3 1 1 2 2 2 2 1 1 1 1 2 1	8 12 8

## Задача G. Очень Легкая Задача

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

Сегодня утром жюри решило добавить в вариант олимпиады еще одну, Очень Легкую Задачу. Ответственный секретарь Оргкомитета напечатал ее условие в одном экземпляре, и теперь ему нужно до начала олимпиады успеть сделать еще  $n$  копий. В его распоряжении имеются два ксерокса, один из которых копирует лист за  $x$  секунд, а другой — за  $y$ . Разрешается использовать как один ксерокс, так и оба одновременно. Можно копировать не только с оригинала, но и с копии. Помогите жюри выяснить, какое минимальное время для этого потребуется.

### Формат входных данных

На вход программы поступают три натуральных числа  $n$ ,  $x$  и  $y$ , разделенные пробелом ( $1 \leq n \leq 2 \cdot 10^8, 1 \leq x, y \leq 10$ ).

### Формат выходных данных

Выведите одно число – минимальное время в секундах, необходимое для получения  $n$  копий.

### Примеры

стандартный ввод	стандартный вывод
4 1 1	3
5 1 2	4



## Задача Н. Провода

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

Дано  $n$  ( $1 \leq n \leq 10^4$ ) отрезков провода длиной  $l_1, l_2, \dots, l_n$  ( $100 \leq l_i \leq 10^7$ ) сантиметров. Требуется с помощью разрезания получить из них  $k$  ( $1 \leq k \leq 10^4$ ) равных отрезков как можно большей длины, выражающейся целым числом сантиметров. Если нельзя получить  $k$  отрезков длиной даже 1 см, вывести 0.

### Формат входных данных

На первой строке заданы числа  $n$  и  $k$ . В следующих  $n$  строках заданы  $l_i$  по одному в строке. Все числа целые.

### Формат выходных данных

Выведите одно число — полученную длину отрезков.

### Пример

стандартный ввод	стандартный вывод
4 11 802 743 457 539	200

## Задача I. Космическое поселение

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

Для освоения Марса требуется построить исследовательскую базу. База должна состоять из  $n$  одинаковых модулей, каждый из которых представляет собой прямоугольник.

Каждый модуль представляет собой жилой отсек, который имеет форму прямоугольника размером  $a \times b$  метров. Для повышения надежности модулей инженеры могут добавить вокруг каждого модуля слой дополнительной защиты. Толщина этого слоя должна составлять целое число метров, и все модули должны иметь одинаковую толщину дополнительной защиты. Модуль с защитой, толщина которой равна  $d$  метрам, будет иметь форму прямоугольника размером  $(a + 2d) \times (b + 2d)$  метров.

Все модули должны быть расположены на заранее подготовленном прямоугольном поле размером  $w \times h$  метров. При этом они должны быть организованы в виде регулярной сетки: их стороны должны быть параллельны сторонам поля, и модули должны быть ориентированы одинаково.

Требуется написать программу, которая по заданному количеству и размеру модулей, а также размеру поля для их размещения, определяет максимальную толщину слоя дополнительной защиты, который можно добавить к каждому модулю.

### Формат входных данных

На вход программы подается пять разделенных пробелами целых чисел:  $n, a, b, w, h$  ( $1 \leq n, a, b, w, h \leq 10^{18}$ ). Гарантируется, что без дополнительной защиты все модули можно разместить в поселении описанным образом.

### Формат выходных данных

Выведите одно целое число: максимальную возможную толщину дополнительной защиты. Если дополнительную защиту установить не удастся, требуется вывести число 0.

### Примеры

стандартный ввод	стандартный вывод
11 2 3 21 25	2
1 5 5 6 6	0

## Задача J. Вырубка леса

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

Фермер Николай нанял двух лесорубов: Дмитрия и Федора, чтобы вырубить лес, на месте которого должно быть кукурузное поле. В лесу растут  $X$  деревьев.

Дмитрий срубает по  $A$  деревьев в день, но каждый  $K$ -й день он отдыхает и не срубает ни одного дерева. Таким образом, Дмитрий отдыхает в  $K$ -й,  $2K$ -й,  $3K$ -й день, и т.д.

Федор срубает по  $B$  деревьев в день, но каждый  $M$ -й день он отдыхает и не срубает ни одного дерева. Таким образом, Федор отдыхает в  $M$ -й,  $2M$ -й,  $3M$ -й день, и т.д.

Лесорубы работают параллельно и, таким образом, в дни, когда никто из них не отдыхает, они срубают  $A + B$  деревьев, в дни, когда отдыхает только Федор —  $A$  деревьев, а в дни, когда отдыхает только Дмитрий —  $B$  деревьев. В дни, когда оба лесоруба отдыхают, ни одно дерево не срубается.

Фермер Николай хочет понять, за сколько дней лесорубы срубят все деревья, и он сможет засеять кукурузное поле.

Требуется написать программу, которая по заданным целым числам  $A$ ,  $K$ ,  $B$ ,  $M$  и  $X$  определяет, за сколько дней все деревья в лесу будут вырублены.

### Формат входных данных

Входной файл содержит пять целых чисел, разделенных пробелами:  $A$ ,  $K$ ,  $B$ ,  $M$  и  $X$  ( $1 \leq A, B \leq 10^9$ ,  $1 \leq K, M \leq 10^{18}$ ,  $1 \leq X \leq 10^{18}$ ). Гарантируется, что  $K$  и  $M$  не будут одновременно равны 1.

### Формат выходных данных

Выходной файл должен содержать одно целое число — искомое количество дней.

### Пример

стандартный ввод	стандартный вывод
2 4 3 3 25	7

## Задача К. К-Best

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

У Демьяны есть  $n$  драгоценностей. Каждая из драгоценностей имеет ценность  $v_i$  и вес  $w_i$ . С тех пор, как её мужа Джонни уволили в связи с последним финансовым кризисом, Демьяна решила продать несколько драгоценностей. Для себя она решила оставить лишь  $k$  лучших. Лучших в смысле максимизации достаточно специфического выражения: пусть она оставила для себя драгоценности номер  $i_1, i_2, \dots, i_k$ , тогда максимальной должна быть величина

$$\frac{\sum_{j=1}^k v_{i_j}}{\sum_{j=1}^k w_{i_j}}$$

Помогите Демьяне выбрать  $k$  драгоценностей требуемым образом.

### Формат входных данных

На первой строке  $n$  и  $k$  ( $1 \leq k \leq n \leq 100\,000$ ).

Следующие  $n$  строк содержат пары целых чисел  $v_i, w_i$  ( $0 \leq v_i \leq 10^6, 1 \leq w_i \leq 10^6$ , сумма всех  $v_i$  не превосходит  $10^7$ , сумма всех  $w_i$  также не превосходит  $10^7$ ).

### Формат выходных данных

Выведите  $k$  различных чисел от 1 до  $n$  — номера драгоценностей. Драгоценности нумеруются в том порядке, в котором перечислены во входных данных. Если есть несколько оптимальных ответов, выведите любой.

### Пример

стандартный ввод	стандартный вывод
3 2 1 1 1 2 1 3	1 2

## Задача L. Одинокое число

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

*Это интерактивная задача.*

Однажды Маше было нечего делать, и она записала на листе бумаги  $N$  целых чисел:  $a_1, a_2, \dots, a_N$ . Маша недавно изучила алгоритмы сортировки, поэтому она выписала свои числа в неубывающем порядке, то есть  $a_1 \leq a_2 \leq \dots \leq a_N$ .

Также Маша очень любит загадки, поэтому среди ее чисел есть некоторое число  $C$ , которое встречается среди выписанных чисел ровно один раз, а все остальные числа встречаются ровно два раза.

Маша загадала вам загадку — найти «одинокое» число  $C$ . Для этого вы можете не более, чем 42 раза попросить Машу сообщить вам  $i$ -е записанное число.

Маша сообщила вам, что  $1 \leq a_i \leq 10^9$ .

### Протокол взаимодействия

В начале ваше решение должно считать число  $N$  ( $1 \leq N \leq 10^6$ ) — количество записанных Машей чисел.

Затем ваше решение может сделать не более 42 запроса. Для того, чтобы сделать запрос, ваше решение должно вывести его в следующем формате: «?  $i$ » (без кавычек,  $1 \leq i \leq N$ ). Ответом на запрос является число  $a_i$ .

Для того, чтобы вывести ответ, ваше решение должно вывести «!  $C$ », после чего немедленно завершить работу.

Вы должны в точности соблюдать протокол взаимодействия с интерактором, в противном случае решение может получить произвольный вердикт.

При превышении числа запросов вы получите вердикт «Неправильный ответ».

После каждого запроса, в том числе после вывода ответа, вы должны выполнить операцию `flush`.

Для сброса буфера вывода (то есть для операции «`flush`») сразу после вывода запроса и перевода строки нужно сделать:

- `fflush(stdout)` или `cout.flush()` в языке C++;
- `System.out.flush()` в Java;
- `sys.stdout.flush()` в Python;
- `flush(output)` в Pascal;
- смотрите документацию для других языков.

Если вы не сделаете операцию `flush` после какого-либо запроса, ваше решение может получить любой вердикт.

## Пример

стандартный ввод	стандартный вывод
5	? 1
1	? 2
1	? 3
4	? 4
5	? 5
5	! 4

## Замечание

В примере записанные числа равны: 1, 1, 4, 5, 5. Одиноким числом, конечно, является число 4.

## Задача М. Осколки прошлого

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

Зло повержено! Тёмный маг Деймос не смог открыть врата в свой ледяной мир и вынужден был с позором бежать. Теперь Дезмонд женится на принцессе Алатиэль, и они будут править Энией.

Разумеется, Торвальд был приглашён на свадьбу. Прогдегустировав изрядное количество эльфийских вин на церемонии, он пошёл погулять по огромному королевскому дворцу, но заблудился. Он шёл только вперёд из комнаты в комнату, но все комнаты были невероятно похожи друг на друга и отличались лишь тем, что в некоторых горели свечи, а в некоторых было темно. Пройдя уже довольно много комнат, Торвальд понял, что ходит по кругу. Дело в том, что дворец имеет круглую форму, а комнаты по периметру дворца соединены друг с другом по циклу: из каждой комнаты можно перейти в следующую или в предыдущую, но если всегда двигаться в одну сторону, то просто вернёшься туда, где уже был.

Торвальду надоело бесцельно ходить по комнатам, и он решил выяснить, сколько их всего. Он не хотел бы трогать что-либо во дворце, чтобы случайно это не повредить. Единственное, что он может делать — зажигать и гасить свечи в комнатах. Кроме того, он находится не в лучшем состоянии духа и скоро заснёт, так что количество операций, которое он сможет выполнить, ограничено. Чем меньше комнат, тем они просторней, поэтому Торвальд дольше будет выполнять в них операции и успеет сделать меньше, прежде чем заснёт.

### Формат входных данных

Это интерактивная задача. Здесь ваша программа должна по мере решения обмениваться информацией с программой жюри. Обратите внимание, что после вывода каждого сообщения ваша программа должна очищать потоковый буфер, чтобы выведенная вами информация дошла до программы жюри: используйте для этого `cout << endl` после каждого вашего вывода.

В начале на вход вашей программе приходит состояние свечей в комнате, в которой изначально находится Торвальд. В случае, если свечи в ней горят, на вход вашей программе будет передано единственное целое число 1 в отдельной строке, иначе — аналогичным образом будет передано число 0.

После этого ваша программа должна сообщать, какую операцию она выполняет. Для этого ей следует выводить один заглавный латинский символ в отдельной строке:

- 'L' — перейти в предыдущую комнату;
- 'R' — перейти в следующую комнату;
- 'S' — изменить состояние свечей в текущей комнате (зажечь их, если они были потушены, или потушить, если они горели).

После каждой операции перехода вам сообщается состояние свечей в комнате, в которую только что перешёл Торвальд.

Если вы уверены, что знаете количество комнат во дворце, то вместо символа, обозначающего операцию, выведите это количество в единственной строке. После этого ваша программа должна завершиться.

Всего во дворце  $n$  комнат ( $1 \leq n \leq 2000$ ). Торвальд должен выяснить правильный ответ, затратив на это не более  $10n$  операций.

## Пример

стандартный ввод	стандартный вывод
1	R
1	S
0	L
	1

## Замечание

Пояснение к примеру. Торвальд находился в комнате с включенными свечами. Он пошёл вправо, там свечи тоже были включены. После этого он потушил свечи, пошёл влево и обнаружил, что в первой комнате свечи тоже погасли. И тут он понял, что периметр дворца состоит из единственной комнаты.

Лишние пустые строки в примере приведены для удобства. При решении задачи вам нужно выводить ровно один символ переноса строки. Программа жюри не будет выводить символы переноса строки.



## Задача N. Oracle соло мид

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

Меха-Наруто играет в компьютерную игру. У его персонажа есть следующая способность: нанести вражескому герою  $a$  единиц урона, затем восполнять ему  $b$  единиц здоровья в конце каждой секунды, начиная со следующей, на протяжении ровно  $c$  секунд. В частности, если эта способность применяется в момент времени  $t$ , то здоровье врага уменьшается на  $a$  в момент времени  $t$ , а затем увеличивается на  $b$  в моменты  $t + 1, t + 2, \dots, t + c$ .

У способности есть время перезарядки, равное  $d$  секундам, то есть если Меха-Наруто применяет способность в момент времени  $t$ , то в следующий раз он может её применить не раньше момента  $t + d$ . По некоторым причинам Меха-Наруто может использовать способность только в целые моменты времени, поэтому все изменения здоровья врага также происходят в целочисленные моменты.

Эффекты от разных применений заклинания накладываются друг на друга. В частности, если вражеский герой находится под действием  $k$  заклинаний, применённых ранее и ещё не истёкших, то его здоровье увеличится на  $k \cdot b$ . Помимо этого все изменения, которые происходят в один и тот же момент времени, учитываются одновременно.

Теперь Меха-Наруто интересно, может ли он убить своего оппонента просто применяя свою способность так часто, как только можно (то есть каждые  $d$  секунд). Герой считается погибшим, если уровень его здоровья становится равным 0 или ниже. Предположим, что здоровье вражеского персонажа не изменяется никаким образом, кроме как от применения заклинания. Какое наибольшее количество здоровья может быть у врага, чтобы Меха-Наруто мог его убить?

### Формат входных данных

Первая строка содержит единственное число  $t$  ( $1 \leq t \leq 10^5$ ) — число тестов.

Каждый тест описывается четвёркой натуральных чисел  $a, b, c$  и  $d$  ( $1 \leq a, b, c, d \leq 10^6$ ), записанных через пробел и означающих соответственно мгновенный урон от способности, ежесекундно восполняемый объём здоровья, время действия каждого заклинания и время перезарядки способности.

### Формат выходных данных

Для каждого теста выведите на отдельной строке  $-1$ , если способность может рано или поздно убить любого врага, каким бы большим ни был его уровень здоровья; в противном случае выведите наибольшее число здоровья, при котором оппонент будет убит.

### Пример

стандартный ввод	стандартный вывод
7	1
1 1 1 1	2
2 2 2 2	1
1 2 3 4	5
4 3 2 1	534
228 21 11 3	-1
239 21 11 3	500000500000
1000000 1 1000000 1	

### Замечание

В первом тесте из условия любая единица урона отменяется через секунду, поэтому Меха-Наруто не может нанести больше, чем 1 единицу урона.

В четвёртом тесте из условия герой оппонента получает:

- 4 урона (1-е применение способности) в момент 0;

- 4 урона (2-е применение способности), и 3 единицы здоровья восполняются (1-е применение) в момент 1 (всего 5 урона к начальному уровню здоровья);
- 4 урона (3-е применение способности), и 6 здоровья восполняется (1-е и 2-е применения) в момент 2 (всего 3 урона к изначальному здоровью);
- и так далее.

Можно доказать, что ни к какому моменту времени враг не получит суммарно 6 или больше урона, поэтому ответ на этот тест есть 5. Обратите внимание, как производится пересчёт здоровья: например, если бы у врага было 8 здоровья, он бы **не** умер в момент времени 1, как если бы мы сначала вычли из его здоровья 4 единицы, а затем сочли бы его мёртвым, не успев добавить 3 единицы от лечения.

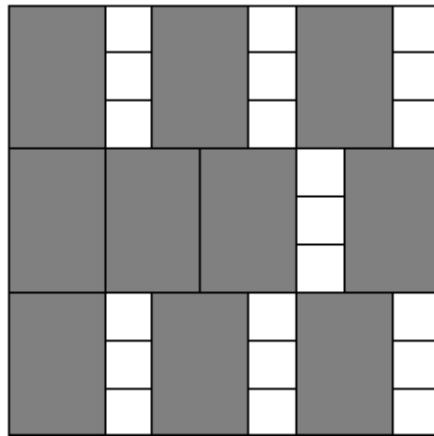
В шестом тесте из условия герой со сколько угодно большим количеством здоровья рано или поздно умрёт.

В седьмом тесте из условия ответ не помещается в 32-битный целочисленный тип.

## Задача О. Дипломы

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

Когда Петя учился в школе, он часто участвовал в олимпиадах по информатике, математике и физике. Так как он был достаточно способным мальчиком и усердно учился, то на многих из этих олимпиад он получал дипломы. К окончанию школы у него накопилось  $n$  дипломов, причём, как оказалось, все они имели одинаковые размеры:  $w$  — в ширину и  $h$  — в высоту. Сейчас Петя учится в одном из лучших российских университетов и живёт в общежитии со своими одногруппниками. Он решил украсить свою комнату, повесив на одну из стен свои дипломы за школьные олимпиады. Так как к бетонной стене прикрепить дипломы достаточно трудно, то он решил купить специальную доску из пробкового дерева, чтобы прикрепить её к стене, а к ней — дипломы. Для того чтобы эта конструкция выглядела более красиво, Петя хочет, чтобы доска была квадратной и занимала как можно меньше места на стене. Каждый диплом должен быть размещён строго в прямоугольнике размером  $w$  на  $h$ . Дипломы запрещается поворачивать на 90 градусов. Прямоугольники, соответствующие различным дипломам, не должны иметь общих внутренних точек. Требуется написать программу, которая вычислит минимальный размер стороны доски, которая потребуется Пете для размещения всех своих дипломов.



### Формат входных данных

Входной файл содержит три целых числа:  $w, h, n$  ( $1 \leq w, h, n \leq 10^9$ ).

### Формат выходных данных

В выходной файл необходимо вывести одно целое число — ответ на поставленную задачу.

### Примеры

стандартный ввод	стандартный вывод
2 3 10	9
1 1 1	1