Задача A. Перевёрнутая цепная дробь

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Конечная цепная дробь — это последовательность вида $[a_0; a_1, a_2, \ldots, a_n]$. На элементы цепной дроби и её размер накладываются следующие ограничения:

- n — неотрицательное конечное целое число,
- элементы a_0, a_1, a_2, …, a_n — целые числа,
- $a_i > 0$ при $i > 0$,
- $a_n > 1$, если $n > 0$.

Эти ограничения позволяют установить взаимно однозначное соответствие между рациональными числами и конечными цепными дробями: каждому рациональному числу x соответствует единственная цепная дробь $[a_0; a_1, a_2, \ldots, a_n]$ такая, что

$$x = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{\ddots + \cfrac{1}{a_n}}}}.$$

Для обозначения соответствия используется знак равенства: $x = [a_0; a_1, a_2, \ldots, a_n]$. Например,

$$\frac{17}{25} = 0 + \frac{1}{\frac{25}{17}} = 0 + \frac{1}{1 + \frac{8}{17}} = 0 + \frac{1}{1 + \frac{1}{\frac{17}{8}}} = \mathbf{0} + \frac{1}{\mathbf{1} + \frac{1}{\mathbf{2} + \frac{1}{\mathbf{8}}}},$$

поэтому мы пишем $\frac{17}{25} = [0; 1, 2, 8]$.

Вам дано рациональное число x ($0 < x \leqslant \frac{1}{2}$). Пусть $x = [0; a_1, a_2, \ldots, a_n]$. Найдите рациональное число, равное $[0; a_n, a_{n-1}, \ldots, a_1]$.

Формат входных данных

В единственной строке даны два положительных целых числа p и q ($1 \leqslant p < q \leqslant 10^9$). Гарантируется, что $\frac{p}{q}$ — несократимая дробь, причём $0 < \frac{p}{q} \leqslant \frac{1}{2}$.

Формат выходных данных

Выведите искомое число в виде несократимой дроби. Числитель и знаменатель дроби следует выводить через пробел.

Примеры

стандартный ввод	**стандартный вывод**
3 7	2 7
2 7	3 7
2 5	2 5

Задача B. Золотое сечение

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Вы, наверное, уже знаете, что цепная дробь — это последовательность вида

$$[a_0; a_1, a_2, \dots, a_n],$$

которой ставится в соответствие число

$$x = a_0 + \frac{1}{a_1 + \frac{1}{\dots + \frac{1}{a_n}}}.$$

Здесь $n \geqslant 0$, a_i — целые числа, $a_i > 0$ при $i > 0$ и $a_n > 1$ при $n > 0$; такие условия нужны для того, чтобы между числами и цепными дробями существовало взаимно однозначное соответствие.

Заметим, что если цепная дробь числа x конечна, то, последовательно приводя дроби к общему знаменателю снизу вверх, её можно свести к виду $\frac{p}{q}$, где p и q — целые числа. Так, например, для $x = [1; 2, 1, 3]$ получаем

$$x = 1 + \frac{1}{2 + \frac{1}{1+\frac{1}{3}}} = 1 + \frac{1}{2 + \frac{3}{4}} = 1 + \frac{4}{11} = \frac{15}{11}.$$

Цепные дроби полезны прежде всего тем, что, если рассматривать лишь первые несколько членов дроби, то соответствующее число будет достаточно хорошим приближением для исходной цепной дроби. Так, в приведённом примере число $y = [1; 2, 1] = [1; 3]$ (вспомним, что в каноническом виде последнее число цепной дроби при $n > 0$ должно быть больше единицы) отличается от x всего на $|\frac{4}{3} - \frac{15}{11}| = \frac{1}{33}$.

В этой задаче мы будем искать хорошие приближения для числа $\phi = \frac{\sqrt{5}+1}{2} \equiv 1.6180339887\dots$ — отношения золотого сечения. Это число известно как отношение частей отрезка, при котором весь отрезок относится к большей части так же, как бóльшая часть к меньшей.

Известно, что цепная дробь ϕ — это бесконечная последовательность $[1; 1, 1, 1, 1, 1, \dots]$. Поиск приближения будет состоять в том, чтобы взять первые n членов этой бесконечной последовательности и найти число, соответствующее полученной цепной дроби, в виде обыкновенной дроби $\frac{p}{q}$. Кроме того, требуется, чтобы эта дробь была несократима, то есть наибольший общий делитель p и q был равен единице.

Формат входных данных

Входной файл содержит одно целое число n ($1 \leqslant n \leqslant 1000$).

Формат выходных данных

Выведите в выходной файл два целых числа: в первой строке выведите p, а во второй — q. Числа не должны начинаться с нуля.

Примеры

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

Пояснения к примерам

В первом примере цепная дробь $[1; 1]$ приводится к каноническому виду $[2]$, и соответствующее приближение оказывается равным двум.

Во втором примере цепная дробь $[1; 1, 1]$ преобразуется к каноническому виду $[1, 2]$, и ответ равен $\frac{3}{2}$.

Задача C. Буквы

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Построим бесконечную строку из букв «a» и «b».

Начнём со строки «a». Зафиксируем число n. Теперь бесконечное количество раз сделаем следующую операцию: возьмём строку, которая у нас есть, и одновременно заменим все буквы «a» на строки «aaa...aab», в которых n букв «a» и одна буква «b», а все буквы «b» на строки «a». Например, при $n = 2$ первые несколько операций выглядят так:

«a» $\longrightarrow$ «aab» $\longrightarrow$ «aabaaba» $\longrightarrow$ «aabaabaaabaabaaab» $\longrightarrow \ldots$

Выведите подстроку полученной бесконечной строки: от ℓ-й буквы до r-й включительно. Буквы в строке нумеруются с единицы.

Формат входных данных

В первой строке заданы три целых числа: n, ℓ и r. Ограничения:

- $1 \leqslant n \leqslant 10$
- $1 \leqslant \ell, r \leqslant 10^{15}$
- $0 \leqslant r - \ell \leqslant 1000$

Формат выходных данных

В первой строке выведите буквы с ℓ-й до r-й без пробелов.

Примеры

стандартный ввод	*стандартный вывод*
2 1 17	aabaabaaabaabaaab
10 104 111	aaaaaaba

Задача D. Сравнение цепных дробей

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

В этой задаче нужно сравнить два рациональных числа, заданных в виде цепных дробей.

Конечная цепная дробь — это последовательность вида $[a_0; a_1, a_2, \ldots, a_n]$. На элементы цепной дроби и её размер накладываются следующие ограничения:

- n — неотрицательное конечное целое число,
- элементы a_0, a_1, a_2, …, a_n — целые числа,
- $a_i > 0$ при $i > 0$,
- $a_n > 1$, если $n > 0$.

Эти ограничения позволяют установить взаимно однозначное соответствие между рациональными числами и конечными цепными дробями: каждому рациональному числу x соответствует единственная цепная дробь $[a_0; a_1, a_2, \ldots, a_n]$ такая, что

$$x = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{\ddots + \cfrac{1}{a_n}}}}.$$

Для обозначения соответствия используется знак равенства: $x = [a_0; a_1, a_2, \ldots, a_n]$. Например,

$$\frac{17}{25} = 0 + \frac{1}{\frac{25}{17}} = 0 + \frac{1}{1 + \frac{8}{17}} = 0 + \frac{1}{1 + \frac{1}{\frac{17}{8}}} = \mathbf{0} + \frac{1}{\mathbf{1} + \frac{1}{\mathbf{2} + \frac{1}{\mathbf{8}}}},$$

поэтому мы пишем $\frac{17}{25} = [0; 1, 2, 8]$.

По двум цепным дробям, задающим рациональные числа x и y, выясните, какое утверждение верно: $x < y$, $x = y$ или $x > y$.

Формат входных данных

Входные данные состоят из двух строк. Первая строка содержит цепную дробь, задающую рациональное число x. Вторая строка содержит цепную дробь, задающую рациональное число y.

Каждая цепная дробь задаётся как последовательность целых чисел, разделённых одиночными пробелами. Описание начинается с целого числа n — длины цепной дроби ($0 \leqslant n \leqslant 100\,000$). Далее следует $(n + 1)$ число — элементы цепной дроби: a_0, a_1, a_2, …, a_n ($|a_i| \leqslant 10^9$). Гарантируется, что $a_i > 0$ при $i > 0$, а кроме того, $a_n > 1$, если $n > 0$.

Формат выходных данных

В первой строке выведите один символ: «`<`», если $x < y$, «`=`», если $x = y$, и «`>`», если $x > y$.

Примеры

стандартный ввод	*стандартный вывод*	Пояснение
`1 0 3` `2 0 1 2`	`<`	$x = 0 + \frac{1}{3} = \frac{1}{3}$, $y = 0 + \frac{1}{1+\frac{1}{2}} = \frac{2}{3}$
`0 1` `0 1`	`=`	$x = 1$, $y = 1$
`1 -1 2` `0 -1`	`>`	$x = -1 + \frac{1}{2} = -\frac{1}{2}$, $y = -1$

Задача E. Умножение на два — лайт

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Конечная цепная дробь — это последовательность вида

$$[a_0; a_1, \ldots, a_n],$$

где $0 \leqslant n < +\infty$, $a_0, a_1, a_2, \ldots, a_n$ — целые числа, $a_i > 0$ при $i > 0$ и $a_n > 1$, если $n > 0$. Цепной дроби ставится в соответствие число

$$x = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{\ddots + \cfrac{1}{a_n}}}}.$$

Известно, что каждому рациональному числу соответствует ровно одна конечная цепная дробь, а каждой конечной цепной дроби соответствует ровно одно рациональное число. Поэтому для обозначения соответствия используется знак равенства: $x = [a_0; a_1, a_2, \ldots, a_n]$.

Дано рациональное число x, записанное в виде цепной дроби. Требуется найти цепную дробь числа $2 \cdot x$.

Формат входных данных

В первой строке задана исходная цепная дробь в виде $[a_0; a_1, a_2, ..., a_n]$. Гарантируется, что все элементы цепной дроби целые, $0 \leqslant n \leqslant 10\,000$, для первого элемента верно $0 \leqslant a_0 \leqslant 100$, для всех $i = 1, 2, \ldots, n$ выполнено $1 \leqslant a_i \leqslant 100$, а кроме того, если $n > 0$, то $a_n \geqslant 2$.

Формат выходных данных

В первой строке выведите полученную цепную дробь в виде $[b_0; b_1, b_2, ..., b_m]$. Все элементы цепной дроби должны быть целыми, для всех $i = 1, 2, \ldots, m$ должно быть выполнено $1 \leqslant b_i \leqslant 100$, а кроме того, если $m > 0$, то $b_m \geqslant 2$. Для цепных дробей должно выполняться равенство

$$2 \cdot [a_0; a_1, a_2, \ldots, a_n] = [b_0; b_1, b_2, \ldots, b_m].$$

Примеры

стандартный ввод	**стандартный вывод**
`[0; 1, 2]`	`[1; 3]`
`[0; 2]`	`[1]`
`[1]`	`[2]`
`[1; 5, 1, 1, 2]`	`[2; 2, 1, 4]`

Задача F. Умножение на два

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Конечная цепная дробь — это последовательность вида

$$[a_0; a_1, \ldots, a_n],$$

где $0 \leqslant n < +\infty$, $a_0, a_1, a_2, \ldots, a_n$ — целые числа, $a_i > 0$ при $i > 0$ и $a_n > 1$, если $n > 0$. Цепной дроби ставится в соответствие число

$$x = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{\ddots + \cfrac{1}{a_n}}}}.$$

Известно, что каждому рациональному числу соответствует ровно одна конечная цепная дробь, а каждой конечной цепной дроби соответствует ровно одно рациональное число. Поэтому для обозначения соответствия используется знак равенства: $x = [a_0; a_1, a_2, \ldots, a_n]$.

Дано рациональное число x, записанное в виде цепной дроби. Требуется найти цепную дробь числа $2 \cdot x$.

Формат входных данных

В первой строке задана исходная цепная дробь в виде $[a_0; a_1, a_2, ..., a_n]$. Гарантируется, что все элементы цепной дроби целые, $0 \leqslant n \leqslant 10^6$, для первого элемента верно $0 \leqslant a_0 \leqslant 100$, для всех $i = 1, 2, \ldots, n$ выполнено $1 \leqslant a_i \leqslant 100$, а кроме того, если $n > 0$, то $a_n \geqslant 2$.

Формат выходных данных

В первой строке выведите полученную цепную дробь в виде $[b_0; b_1, b_2, ..., b_m]$. Все элементы цепной дроби должны быть целыми, для всех $i = 1, 2, \ldots, m$ должно быть выполнено $1 \leqslant b_i \leqslant 100$, а кроме того, если $m > 0$, то $b_m \geqslant 2$. Для цепных дробей должно выполняться равенство

$$2 \cdot [a_0; a_1, a_2, \ldots, a_n] = [b_0; b_1, b_2, \ldots, b_m].$$

Примеры

стандартный ввод	стандартный вывод
`[0; 1, 2]`	`[1; 3]`
`[0; 2]`	`[1]`
`[1]`	`[2]`
`[1; 5, 1, 1, 2]`	`[2; 2, 1, 4]`

Задача G. Противоположная цепная дробь — лайт

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Конечная цепная дробь — это последовательность вида

$$[a_0; a_1, \ldots, a_n],$$

где $0 \leqslant n < +\infty$, $a_0, a_1, a_2, \ldots, a_n$ — целые числа, $a_i > 0$ при $i > 0$ и $a_n > 1$, если $n > 0$. Цепной дроби ставится в соответствие число

$$x = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{\ddots + \cfrac{1}{a_n}}}}.$$

Известно, что каждому рациональному числу соответствует ровно одна конечная цепная дробь, а каждой конечной цепной дроби соответствует ровно одно рациональное число. Поэтому для обозначения соответствия используется знак равенства: $x = [a_0; a_1, a_2, \ldots, a_n]$.

Дано рациональное число x $(0 < x < 1)$, записанное в виде цепной дроби. Требуется найти цепную дробь числа $(1 - x)$.

Формат входных данных

В первой строке задано целое число n $(1 \leqslant n \leqslant 10\,000)$. Во второй строке дано $(n+1)$ целых чисел: a_0, a_1, a_2, …, a_n — элементы исходной цепной дроби. Гарантируется, что цепная дробь задана корректно, то есть заданные числа удовлетворяют всем ограничениям из определения. Заданная цепная дробь соответствует такому рациональному числу x, что $0 < x < 1$, и для каждого i выполнено неравенство $a_i \leqslant 100$.

Формат выходных данных

В первой строке выведите целое число m, а во второй строке — $(m + 1)$ положительных целых чисел: b_0, b_1, b_2, …, b_m — элементы полученной цепной дроби. Цепная дробь должна быть задана корректно, то есть выведенные числа должны удовлетворять всем ограничениям из определения. Для цепных дробей должно выполняться равенство

$$1 - [a_0; a_1, a_2, \ldots, a_n] = [b_0; b_1, b_2, \ldots, b_m].$$

Примеры

стандартный ввод	стандартный вывод
2 0 1 2	1 0 3
2 0 2 3	3 0 1 1 3

Задача H. Противоположная цепная дробь

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Конечная цепная дробь — это последовательность вида

$$[a_0; a_1, \ldots, a_n],$$

где $0 \leqslant n < +\infty$, $a_0, a_1, a_2, \ldots, a_n$ — целые числа, $a_i > 0$ при $i > 0$ и $a_n > 1$, если $n > 0$. Цепной дроби ставится в соответствие число

$$x = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{\ddots + \cfrac{1}{a_n}}}}.$$

Известно, что каждому рациональному числу соответствует ровно одна конечная цепная дробь, а каждой конечной цепной дроби соответствует ровно одно рациональное число. Поэтому для обозначения соответствия используется знак равенства: $x = [a_0; a_1, a_2, \ldots, a_n]$.

Дано рациональное число x $(0 < x < 1)$, записанное в виде цепной дроби. Требуется найти цепную дробь числа $(1 - x)$.

Формат входных данных

В первой строке задано целое число n $(1 \leqslant n \leqslant 10^6)$. Во второй строке дано $(n + 1)$ целых чисел: a_0, a_1, a_2, …, a_n — элементы исходной цепной дроби. Гарантируется, что цепная дробь задана корректно, то есть заданные числа удовлетворяют всем ограничениям из определения. Заданная цепная дробь соответствует такому рациональному числу x, что $0 < x < 1$, и для каждого i выполнено неравенство $a_i \leqslant 100$.

Формат выходных данных

В первой строке выведите целое число m, а во второй строке — $(m + 1)$ положительных целых чисел: b_0, b_1, b_2, …, b_m — элементы полученной цепной дроби. Цепная дробь должна быть задана корректно, то есть выведенные числа должны удовлетворять всем ограничениям из определения. Для цепных дробей должно выполняться равенство

$$1 - [a_0; a_1, a_2, \ldots, a_n] = [b_0; b_1, b_2, \ldots, b_m].$$

Примеры

стандартный ввод	**стандартный вывод**
2 0 1 2	1 0 3
2 0 2 3	3 0 1 1 3

Задача I. Цепная дробь для корня — лайт

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Цепной дробью называется выражение вида

$$a_0 + \cfrac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\ldots + \frac{1}{a_m}}}},$$

где все a_k целые, $a_k > 0$ при $k > 0$ и $a_m > 1$, если $m > 0$. Эти, на первый взгляд странные, правила позволяют каждому вещественному числу x единственным образом сопоставить цепную дробь так, чтобы выполнялось равенство

$$x = a_0 + \cfrac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\ldots + \frac{1}{a_m}}}}.$$

Например, цепная дробь для числа $7/18$ выглядит как

$$0 + \cfrac{1}{2 + \frac{1}{1 + \frac{1}{1 + \frac{1}{3}}}}:$$

$$\begin{aligned} 7/18 &= 0 + 7/18 \\ &= 0 + 1/(18/7) \\ &= 0 + 1/(2 + 4/7) \\ &= 0 + 1/(2 + 1/(7/4)) \\ &= 0 + 1/(2 + 1/(1 + 3/4)) \\ &= 0 + 1/(2 + 1/(1 + 1/(4/3))) \\ &= 0 + 1/(2 + 1/(1 + 1/(1 + 1/3))). \end{aligned}$$

Цепную дробь можно записывать как $[\mathtt{a}_0; \mathtt{a}_1, \mathtt{a}_2, \ldots, \mathtt{a}_\mathtt{m}]$; в этой записи $7/18 = [0; 2, 1, 1, 3]$.

Не все числа имеют конечную цепную дробь; так, для числа $\sqrt{2} \approx 1,41421356\ldots$ цепная дробь имеет вид $[1; 2, 2, 2, 2, 2, 2, \ldots]$ — воспользовавшись равенством $\sqrt{2} - 1 = 1/(\sqrt{2} + 1)$, получаем:

$$\begin{aligned} \sqrt{2} &= 1 + (\sqrt{2} - 1) \\ &= 1 + 1/(\sqrt{2} + 1) \\ &= 1 + 1/(2 + (\sqrt{2} - 1)) \\ &= 1 + 1/(2 + 1/(\sqrt{2} + 1)) \\ &= \ldots \end{aligned}$$

Цепная дробь такого вида называется *периодической*; в данном случае предпериод — это число $a_0 = 1$, а период — повторяющееся бесконечное количество раз число 2. Для удобства записи период цепной дроби можно записывать в круглых скобках; для $\sqrt{2}$ сокращённая запись будет выглядеть как $[1; (2)]$. Можно доказать, что для всякого целого $N \geqslant 0$ цепная дробь числа $\sqrt{N}$ является конечной, если $\sqrt{N}$ — целое число, и периодической в противном случае.

По данному числу N выведите цепную дробь числа $\sqrt{N}$ в сокращённой записи.

Формат входных данных

В первой строке записано целое число N ($1 \leqslant N \leqslant 1000$).

Формат выходных данных

В первую и единственную строку выведите сокращённую запись цепной дроби числа $\sqrt{N}$. Необходимо как можно более точно соблюдать такой же формат вывода, как в примерах. Гарантируется, что правильный ответ на каждый тест содержит не более 10 000 символов.

Примеры

стандартный ввод	**стандартный вывод**
`1`	`[1]`
`2`	`[1;(2)]`
`3`	`[1;(1,2)]`
`4`	`[2]`
`5`	`[2;(4)]`
`6`	`[2;(2,4)]`
`7`	`[2;(1,1,1,4)]`
`76`	`[8;(1,2,1,1,5,4,5,1,1,2,1,16)]`

Задача J. Цепная дробь для корня

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Цепной дробью называется выражение вида

$$a_0 + \cfrac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\dots + \frac{1}{a_m}}}},$$

где все a_k целые, $a_k > 0$ при $k > 0$ и $a_m > 1$, если $m > 0$. Эти, на первый взгляд странные, правила позволяют каждому вещественному числу x единственным образом сопоставить цепную дробь так, чтобы выполнялось равенство

$$x = a_0 + \cfrac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\dots + \frac{1}{a_m}}}}.$$

Например, цепная дробь для числа $7/18$ выглядит как

$$0 + \cfrac{1}{2 + \frac{1}{1 + \frac{1}{1 + \frac{1}{3}}}}:$$

$$\begin{aligned}
7/18 &= 0 + 7/18 \\
&= 0 + 1/(18/7) \\
&= 0 + 1/(2 + 4/7) \\
&= 0 + 1/(2 + 1/(7/4)) \\
&= 0 + 1/(2 + 1/(1 + 3/4)) \\
&= 0 + 1/(2 + 1/(1 + 1/(4/3))) \\
&= 0 + 1/(2 + 1/(1 + 1/(1 + 1/3))).
\end{aligned}$$

Цепную дробь можно записывать как $[a_0; a_1, a_2, \dots, a_m]$; в этой записи $7/18 = [0; 2, 1, 1, 3]$.

Не все числа имеют конечную цепную дробь; так, для числа $\sqrt{2} \approx 1,41421356\dots$ цепная дробь имеет вид $[1; 2, 2, 2, 2, 2, 2, \dots]$ — воспользовавшись равенством $\sqrt{2} - 1 = 1/(\sqrt{2} + 1)$, получаем:

$$\begin{aligned}
\sqrt{2} &= 1 + (\sqrt{2} - 1) \\
&= 1 + 1/(\sqrt{2} + 1) \\
&= 1 + 1/(2 + (\sqrt{2} - 1)) \\
&= 1 + 1/(2 + 1/(\sqrt{2} + 1)) \\
&= \dots
\end{aligned}$$

Цепная дробь такого вида называется *периодической*; в данном случае предпериод — это число $a_0 = 1$, а период — повторяющееся бесконечное количество раз число 2. Для удобства записи период цепной дроби можно записывать в круглых скобках; для $\sqrt{2}$ сокращённая запись будет выглядеть как $[1; (2)]$. Можно доказать, что для всякого целого $N \geqslant 0$ цепная дробь числа $\sqrt{N}$ является конечной, если $\sqrt{N}$ — целое число, и периодической в противном случае.

По данному числу N выведите цепную дробь числа $\sqrt{N}$ в сокращённой записи.

Формат входных данных

В первой строке записано целое число N ($1 \leqslant N \leqslant 10^5$).

Формат выходных данных

В первую и единственную строку выведите сокращённую запись цепной дроби числа $\sqrt{N}$. Необходимо как можно более точно соблюдать такой же формат вывода, как в примерах. Гарантируется, что правильный ответ на каждый тест содержит не более 10^6 символов.

Примеры

стандартный ввод	**стандартный вывод**
`1`	`[1]`
`2`	`[1;(2)]`
`3`	`[1;(1,2)]`
`4`	`[2]`
`5`	`[2;(4)]`
`6`	`[2;(2,4)]`
`7`	`[2;(1,1,1,4)]`
`76`	`[8;(1,2,1,1,5,4,5,1,1,2,1,16)]`