Системы счисления.

В задачах с двоичной системой счисления запрещается пользоваться операцией возведения в степень.

A. *Перевод двоичной дроби в десятичную*

Переведите число из двоичной системы счисления в десятичную.
Дано число, представленное в виде двоичной дроби: запись длиной не более 30 символов, содержащая цифры 0 и 1 и, возможно, одну точку.
Необходимо вывести данное число в виде десятичной дроби. Выведенный ответ должен отличаться от истинного не более, чем на 10^{-12}.

Input	Output
`0.11`	`0.75`
`11.111`	`3.875`

B. *Перевод десятичной дроби в двоичную*

Переведите десятичное число в двоичную систему.
Дано действительное неотрицательное число, не превосходящее 100, записанное в десятичном виде. Целые числа при этом могут не содержать точку.
Необходимо представить число в виде двоичной дроби с фиксированной точкой и вывести это представление. Ответ должен отличаться от правильного не более, чем на 2^{-32}.

Input	Output
`0.5`	`0.1`
`0.1`	`0.0001100110011001100110011001100110011001100110`

C. *Перевод двоичной периодической дроби в десятичную*

Дана запись целого двоичного числа или двоичной периодической дроби, которая включает в себя:

- Обязательную целую часть.
- Обязательный символ точки, отделяющий целую часть от дробной (если дробная часть существует).
- Необязательную дробную непериодическую часть.
- Необязательную периодическую дробную часть, записываемую в круглых скобках.

Необходимо определить значение этой дроби, сохранить его в переменной типа `float`, при этом выведенное значение должно отличаться от истинного не более, чем на 10^{-12}. Общая длина входной строки не превосходит 30 символов.
(!) Подумайте сколько двоичных разрядов необходимо взять для получения требуемой точности, выраженной в количестве десятичных знаков.

Input	Output
`0.1`	`0.5`
`0.(01)`	`0.33333333333333331483`

D* *Перевод рационального числа в периодическую двоичную дробь*

Дано рациональное число. Запишите его в виде двоичной периодической дроби.
На вход программа получает два натуральных числа N и M ($M, N \leqslant 4 \cdot 10^5$).
Программа должна вывести значение N/M, записанное в виде двоичной периодической дроби, при этом длина непериодической дробной части и длина периода должны быть минимально возможными. Если данное число является конечной двоичной дробью, периодическую часть выводить не надо. Решение должно иметь сложность $O(P)$, где P — длина периода дроби.

Input	Output
`1 2`	`0.1`
`1 3`	`0.(01)`

E. *Перевод двоичной периодической дроби в рациональное число*

Дана запись двоичной дроби, как в задаче `C`. В целых числах точка не ставится. Необходимо представить её в виде несократимой дроби N/M.
Программа должна вывести значения N и M.

Input	Output
`0.1`	`1 2`
`0.(001)`	`1 7`
`0.10(01)`	`7 12`

F* *Перевод рационального числа в периодическую десятичную дробь*

Напишите программу, которая переводит правильную дробь в десятичную, выделив (если нужно), период дроби.

Входная строка содержит два разделённых пробелами числа M и N ($M < N \leqslant 3 \cdot 10^5$).

Программа должна вывести десятичную запись дроби, выделив (если нужно) её период. В качестве разделителя целой и дробной части используйте точку.

Решение должно иметь сложность $O(|P| + |Pr|)$, где $|P|$ — длина периода дроби, $|Pr|$ — длина предпериода дроби.

Input	Output
`1 2`	`0.5`
`3 14`	`0.2(142857)`

G. *Перевод из любой системы в любую*

Напишите программу, переводящую запись числа между двумя произвольными системами счисления.

На вход программа получает три величины: N, A, K, где N и K — натуральные числа, основания системы счисления $(2 \leqslant N, K \leqslant 36)$, A — целое число, записанное в системе счисления с основанием $N, 0 \leqslant A < 2^{31}$.

Необходимо вывести значение A в системе счисления с основанием K без лидирующих нулей.

Input	Output

`2`
`101111`
`16`
	`2F`

`10`
`35`
`36`
	`Z`

H. *Произвольная СС: инкремент*

Первая строка входных данных содержит последовательность символов '0', ..., '9', 'A', ..., 'Z', являющейся записью некоторого неотрицательного числа в системе счисления с основанием `base`. Длина числа не превосходит 100000 символов. Вторая строка входных данных содержит основание системы счисления `base`, не превосходящее 36.

Увеличьте это число на 1 и выведите результат в той же системе счисления.

Input	Output
`19A` `11`	`1A0`

I. *Произвольная СС: декремент*

Первая строка входных данных содержит последовательность символов '0', ..., '9', 'A', ..., 'Z', являющейся записью некоторого неотрицательного числа в системе счисления с основанием `base`. Длина числа не превосходит 100000 символов. Вторая строка входных данных содержит основание системы счисления `base`, не превосходящее 36.

Уменьшите это число на 1 и выведите результат в той же системе счисления.

Input	Output
`1A0` `11`	`19A`

J. *Произвольная СС: сложение*

Первая и вторая строки входных данных содержат последовательности символов '0', ..., '9', 'A', ..., 'Z', являющиеся записью некоторых неотрицательных чисел в системе счисления с основанием `base`. Длина чисел не превосходит 50000 символов. Третья строка входных данных содержит основание системы счисления `base`, не превосходящее 36. Гарантируется, что первые две строки являются корректной записью чисел в системе счисления с основанием `base`.

Выведите результат сложения данных чисел в той же системе счисления.

Input	Output

`2`
`2`
`3`
	`11`

`AAA`
`1`
`11`
	`1000`

`245DSF`
`1010101Z`
`36`
	`10346DUE`

Рассмотрим последовательность Фибоначчи: $F_1 = 1, F_2 = 2$ и $F_n = F_{n-1} + F_{n-2}$ при $n > 2$. Первые элементы этой последовательности таковы: $F_2 = 2, F_3 = 3, F_4 = 5, F_5 = 8$ и т.д.

Любое натуральное число можно представить в виде суммы нескольких членов последовательности Фибоначчи. Такое представление будет неоднозначным (приведите пример). Однако, если потребовать, чтобы в представлении не было двух соседних членов последовательности Фибоначчи, то представление становится единственным (докажите!).

Будем говорить, что число A представимо в фибоначчиевой системе счисления (ФСС) в виде $a_k a_{k-1} \ldots a_1$, где $a_i \in \{0; 1\}$, если $A = a_k \cdot F_k + \cdots + a_1 \cdot F_1$ и в записи $a_k a_{k-1} \ldots a_1$ нет двух единиц подряд.

Вот как записываются небольшие числа в фибоначчиевой системе счисления:

Десятичная	0	1	2	3	4	5	6	7	8	9	10	11	12
Фибоначчиева	0	1	10	100	101	1000	1001	1010	10000	10001	10010	10100	10101

K. *Перевод из ФСС (фибоначчиевой СС) в десятичную*

Дана запись числа в фибоначчиевой системе счисления. Запишите его в десятичной системе счисления.

Программа получает на вход строку из символов 0 и 1 и должна вывести одно целое число. Гарантируется, что результат принимает значения от 0 до $2 \cdot 10^9$.

Input	Output
`10101`	`12`

L. *Перевод из десятичной в ФСС*

Дано целое число от 0 до $2 \cdot 10^9$. Выведите его представление в фибоначчиевой системе счисления без лидирующих нулей.

Input	Output
`12`	`10101`

M. *ФСС: инкремент*

Дано целое неотрицательное число N, записанное в фибоначчиевой системе счисления, длина числа не превосходит 10^5 символов. Выведите значение числа $N + 1$ в фибоначчиевой системе счисления.

Input	Output
`100101`	`101000`

N. *ФСС: декремент*

Дано натуральное число N, записанное в фибоначчиевой системе счисления, длина числа не превосходит 10^5 символов. Выведите значение числа $N - 1$ в фибоначчиевой системе счисления.

Input	Output
`101000`	`100101`

O. *ФСС: сложение*

Даны два числа, записанные в фибоначчиевой системе счисления, длины чисел не превосходят 100000 символов. Выведите значение их суммы в фибоначчиевой системе счисления.

Input	Output
`10010` `10101`	`1000001`

В троичной сбалансированной системе счисления (ТССС) используется основание 3 и три цифры: $0, 1$ и -1. Цифру -1 будем обозначать знаком \$. К достоинствам сбалансированной троичной системы счисления относят простоту хранения отрицательных чисел и удобство нахождения числа, противоположного данному.

Такая система счисления естественным образом возникает в задаче о минимальном количестве целочисленных гирек, при помощи которых можно взвесить что-то на чашечных весах, при этом гирьки можно класть на обе чашки весов.

Вот как записываются небольшие числа в сбалансированной троичной системе счисления:

Десятичная	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9
ТВСС	\$00	\$01	\$1\$	\$10	\$11	\$\$	\$0	\$1	\$	0	1	1\$	10	11	1\$\$	1\$0	1\$1	10\$	100

В русскоязычной литературе встречаются также названия “взвешенная”, “симметричная”. В англоязычной литературе принят термин “balanced”.

P. *Перевод из ТССС в десятичную*

Дана запись числа в сбалансированной троичной системе счисления. Переведите его в десятичную. Гарантируется, что ответ не превосходит по модулю 10^9.

Input	Output
`$01`	`-8`

Q. *Перевод из десятичной в ТССС*

Дано целое число от $-2 \cdot 10^9$ до $2 \cdot 10^9$. Выведите его представление в сбалансированной троичной системе счисления без лидирующих нулей.

Input	Output
`-8`	`$01`

R. *ТССС: инкремент*

Дана запись некоторого числа в сбалансированной троичной системе счисления, длина записи не превосходит 10^5 символов.

Увеличьте это число на 1 и выведите его значение в той же системе.

Input	Output
`$01`	`$1$`

S. *ТССС: декремент*

Дана запись некоторого числа в сбалансированной троичной системе счисления, длина записи не превосходит 10^5 символов.

Уменьшите это число на 1 и выведите его значение в той же системе.

Input	Output
`$1$`	`$01`

T. *ТССС: сложение*

Дано два числа, записанных в сбалансированной троичной системе счисления. Выведите их сумму без лидирующих нулей. Длины входных чисел не превосходят 10^5 символов.

Input	Output
`1$$$` `$0$`	`11`

Пояснение: приведённый пример соответствует выражению $14 + (-10) = 4$.

U. *Красивые числа*

Требуется по заданным числам N и K найти такое D, чтобы число N в системе счисления с основанием D заканчивалось как можно большим количеством цифр K.

Вводятся два целых десятичных числа N и K ($1 \leqslant N \leqslant 10^{11}; 0 \leqslant K \leqslant 9$).

Выведите два числа: d — искомое основание системы счисления и L — количество цифр K, которым заканчивается запись числа N в этой системе счисления. Если искомых D несколько, выведите любое из них, не превосходящее 10^{12} (такое всегда существует).

Input	Output
`49 1`	`3 2`
`7 5`	`3 0`
`4 4`	`5 1`
`9 9`	`10 1`

V. *Дырявое множество*

Определим множества $K[i]$ рекуррентно. Пусть $K[0] = [0, 1]$. Разделим сегмент $[0, 1]$ на три части точками $\frac{1}{3}$ и $\frac{2}{3}$ и удалим из него интервал $\left(\frac{1}{3}, \frac{2}{3}\right)$. Получим множество K[1], состоящее из двух оставшихся сегментов $\left[0, \frac{1}{3}\right]$ и $\left[\frac{2}{3}, 1\right]$.

Каждый из них разделим на три части (точками $\frac{1}{9}$ и $\frac{2}{9}$ для первого сегмента, и точками $\frac{7}{9}$ и $\frac{8}{9}$ — для второго) и удалим средние интервалы $\left(\frac{1}{9}, \frac{2}{9}\right)$ и $\left(\frac{7}{9}, \frac{8}{9}\right)$. Таким образом получаем множество K[2], и т.д.

Пусть мы построили множество $K[i]$. Поделим каждый оставшийся сегмент из $K[i]$ на 3 части и удалим из этих сегментов средние интервалы. Получим, таким образом, из $K[i]$ множество $K[i + 1]$.

Вводятся 3 целых числа n, a, b ($n \leqslant 10^6, a \leqslant b \leqslant 10^{18}$). Необходимо определить, принадлежит ли точка с координатой $\frac{a}{b}$ множеству $K[n]$.

Input	Output
`1 2 4`	`NO`
`2 13 18`	`YES`

W. *Круглые факториалы*

Требуется по заданным числам N и K найти количество нулей в конце записи в системе счисления с основанием K числа $N! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot (N-1) \cdot N$.

В первой строке входных данных содержатся числа N и K, разделённые пробелом, $1 \leqslant N \leqslant 10^9, 2 \leqslant K \leqslant 1000$.

Выведите число X — количество нулей в конце записи числа $N!$ в системе счисления с основанием K.

Input	Output
5 10	1
1 2	0
100 10	24
1000 10	249