Задача A. Взлом хеш-функции

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

В некоторых задачах защиты информации используются так называемые *хеш-функции*. Одним из важнейших классов хеш-функций являются так называемые *полиномиальные хеш-функции*.

Пусть дана строка $S = s_1 s_2 \ldots s_l$, состоящая из цифр от 0 до 9. Тогда значение *полиномиальной хеш-функции* $p(S, x, m)$ вычисляется следующим образом

$$p(S, x, m) = (\sum_{i=1}^{l} s_i x^{i-1}) \bmod m$$

($a \bmod b$ обозначает остаток от деления числа a на число b). Например, пусть $S = 0123$, тогда $p(S, 2, 5) = (0 \cdot 1 + 1 \cdot 2 + 2 \cdot 4 + 3 \cdot 8) \bmod 5 = 4$.

Одним из способов применения хеш-функций является хранение паролей. Часто бывает так, что пароли приходится хранить в незащищенной таблице базы данных, поэтому вместо них хранят хеш-функции от них. При проверке пароля вычисляется хеш-функция от введенной строки и сравнивается со значением, хранящимся в таблице.

Ваша задача состоит в том, чтобы по заданным числам x, m, L и v найти строку S из цифр от 0 до 9 длины L, значение полиномиальной хеш-функции $p(S, x, m)$ равно v.

Формат входного файла

Входной файл содержит четыре целых числа: x (x — простое число, $5 \le x \le 100$), m (m является степенью двойки, $1 \le m \le 256$), L ($10 \le L \le 100$) и v ($0 \le v \le m - 1$).

Формат выходного файла

В выходной файл выведите искомую строку или `NO SOLUTION`, если такой строки не существует.

Примеры

crack.in	crack.out
5 16 10 9	0422207956

Задача B. Минимальный лексикографически циклический сдвиг перестановки

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

Перестановкой порядка n называется последовательность из попарно различных целых положительных чисел p_1, p_2, $\ldots$, p_n, где каждое $1 \le p_i \le n$. Будем говорить, что перестановка q_1, q_2, $\ldots$, q_n *лексикографически меньше* перестановки p_1, p_2, $\ldots$, p_n, если существует такое i, что $q_i < p_i$, а для любого $j < i$ $p_j = q_j$.

Циклическим сдвигом на k перестановки p_1, p_2, $\ldots$, p_n называется последовательность, p_{k+1}, p_{k+2}, $\ldots$, p_n, p_1, $\ldots$, p_k. Отметим, что любой циклический сдвиг перестановки также является перестановкой.

Ваша задача состоит в том, чтобы найти наименьший лексикографически циклический сдвиг заданной перестановки.

Формат входного файла

Первая строка входного файла содержит порядок n ($1 \le n \le 100000$) заданной перестановки. Вторая строка содержит числа p_1, p_2, $\ldots$, p_n, отделенные друг от друга пробелами.

Формат выходного файла

В выходной файл выведите перестановку, являющуюся наименьшим лексикографически циклическим сдвигом перестановки, заданной во входном файле. Используйте такой же формат, в каком перестановка задана во второй строке входного файла.

Примеры

minlex.in	minlex.out
3 3 2 1	1 3 2

Задача C. Плейлист

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

Антон — большой любитель музыки. Но в разные моменты ему нравится разная музыка. Однажды он решил составить рейтинг своих песен, причем считать его так — если песня дослушана до конца, то ее рейтинг увеличивается на единицу, если же он так и не дослушал ее, переключив на следующую, то уменьшается на единицу. Отметим, что вследствие этого рейтинг может стать отрицательным.

Он попросил Вас написать ему программу, которая подсчитывала бы такой рейтинг. Для этого он записал в каком порядке прослушивались песни и в какие моменты времени он переключал песню на следующую. Изначально рейтинги всех песен равны нулю.

Формат входного файла

Первая строка входного файла содержит одно число n ($1 \le n \le 1000$) — количество песен у Антона. В следующих n строках следуют описания песен — название песни, состоящее не более, чем из 50 маленьких латинских букв, и длина песни в секундах, разделенные пробелом. Каждая песня длится положительное число секунд и не более 30 минут. $(n+2)$-я строка содержит два числа m, k ($0 \le k < m \le 1000$) — количество прослушанных песен и количество переключений на следующую. В следующих m строках следуют названия прослушанных песен в порядке их прослушивания (все эти песни — из списка). И, наконец. в последней строке — k целых неотрицательных чисел — последовательные (в неубывающем порядке) времена переключения на следующую песню (гарантируется, что она всегда существует), отсчитываемые в секундах от начала прослушивания первой. Считается, что если Антон переключил песню в момент окончания какой-нибудь, то он пропускает уже следующую.

Формат выходного файла

В выходной файл выведите n чисел — рейтинги песен в том порядке, в котором они даны во входном файле.

Примеры

music.in	music.out
2 songa 10 songb 20 5 2 songa songb songb songa songa 40 55	1 0

Задача D. Апельсины

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

Катя решила пригласить к себе в гости n друзей. Так как ее друзья очень любят фрукты, то в качестве угощения для них она купила m одинаковых апельсинов.

Она хочет разрезать каждый апельсин на одинаковое число равных долек так, чтобы их можно было распределить между гостями (сама Катя апельсины есть не будет), и всем гостям досталось поровну долек.

Напишите программу, которая вычисляет минимальное количество долек, на которое необходимо разрезать каждый апельсин, чтобы были выполнены указанные выше условия.

Формат входного файла

Входной файл содержит два положительных целых числа n и m ($1 \le n, m \le 10^9$).

Формат выходного файла

В выходной файл выведите ответ на задачу.

Примеры

orange.in	orange.out
2 5	2
2 4	1

Задача E. Произведение цифр

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

Найдите количество чисел из отрезка $[l, r]$, которые делятся на произведение своих цифр.

Формат входного файла

Входной файл содержит два целых числа: l и r ($1 \le l \le r \le 10^9$, $|r - l| \le 10^5$).

Формат выходного файла

В выходной файл выведите ответ на задачу.

Примеры

product.in	product.out
1 12	11

Задача F. Точки и отрезки

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

Дано n отрезков на числовой прямой и m точек на этой же прямой. Для каждой из данных точек определите, скольким отрезкам она принадлежит. Точка x считается принадлежащей отрезку с концами a и b, если выполняется двойное неравенство $min(a, b) \le x \le max(a, b)$.

Формат входного файла

Первая строка содержит два целых числа n $(1 \le n \le 10^5)$ — число отрезков и m $(1 \le m \le 10^5)$ — число точек. В следующих n строках по два целых числа a_i и b_i — координаты концов соответствующего отрезка. В последней строке m целых чисел — координаты точек. Все числа во входном файле не превосходят по модулю 10^9.

Формат выходного файла

В выходной файл выведите m чисел — для каждой точки количество отрезков, в которых она содержится.

Примеры

segments.in	segments.out
3 2 0 5 -3 2 7 10 1 6	2 0
1 3 -10 10 -100 100 0	0 0 1

Задача G. Дана строка...

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

Назовем *качеством* строки разность между максимальным и минимальным номерами в алфавите букв, входящих в строку. Например, качество строки `ab` равно $2 - 1 = 1$, а строки `abcz` равно $26 - 1 = 25$.

Дана строка s. Необходимо найти непустую подстроку этой строки, обладающую максимальным качеством, а из всех таких — минимальную по длине.

Формат входного файла

Входной файл содержит непустую строку s, состоящую из строчных букв латинского алфавита. Ее длина не превосходит $2 \cdot 10^5$ символов.

Формат выходного файла

В выходной файл выведите искомую подстроку. Если вариантов ответа несколько, выведите любой.

Примеры

string.in	string.out
aba	ab
zzz	z

Задача H. Сумма

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

Задано целое положительное число x. Найдите число способов представить его в виде суммы четырех положительных целых чисел $x = a + b + c + d$, где $a \le b \le c \le d$.

Формат входного файла

Первая строка входного файла содержит целое число x ($1 \le x \le 1500$).

Формат выходного файла

В выходной файл выведите ответ на задачу.

Примеры

sum.in	sum.out
3	0
5	1