

Problem C. Без сдачи

Input file: *standard input*
Output file: *standard output*
Time limit: 1 секунда
Memory limit: 256 мегабайт

Все периодически сталкиваются с ситуацией, когда у кассира нет сдачи и приходится искать по карманам мелочь, чтобы набрать в точности нужную сумму. Чтобы избежать тех случаев, когда несмотря на все усилия, собрать требуемую сумму так и не удастся, можно всегда иметь при себе такой набор денежных знаков разного номинала, чтобы из него точно можно было набрать любую сумму (разумеется, до какого-то разумного предела). При этом, конечно, желательно, чтобы общее количество денежных знаков было минимально возможным.

Зная все возможные номиналы, Вам нужно определить, какое минимальное количество денежных знаков необходимо для того, чтобы набрать из них любую сумму от 1 до некоторого заданного числа включительно.

Input

В первой строке одно целое положительное число N — количество возможных номиналов денежных единиц, $1 \leq N \leq 1000$. Во второй строке N различных целых положительных чисел через пробел — возможные номиналы. Каждое из чисел не превышает 10^6 . Гарантируется, что одно из этих чисел — единица.

В третьей строке одно целое положительное число S , $1 \leq S \leq 10^{10}$.

Output

В первой и единственной строке одно целое положительное число — минимальное количество денежных знаков доступных номиналов, достаточное для того, чтобы набрать любую сумму от 1 до S включительно.

Example

standard input	standard output
4 1 3 4 7 7	4

Problem D. Интересное число

Input file: *standard input*
Output file: *standard output*
Time limit: 2 секунды
Memory limit: 256 мегабайт

- 13.69! Какое интересное число! — заявил Боб, глядя на кассовый чек.
— Да ну? — ответила Алиса. — И чем же так интересно число 1369?
— 13 в квадрате будет 169, и оба числа можно сложить из цифр числа 1369, — объяснил Боб.
— Подумаешь... - ответила Алиса. — 31 в квадрате равно 961, 19 в квадрате — 361, 6 в квадрате — 36, 3 в квадрате — 9, 1 в квадрате — 1. При этом все эти числа складываются из цифр числа 1369.
— Тогда число 1369 становится ещё более интересным!

Напишите программу, которая в заданном числе находит все «спрятанные» пары чисел и их квадратов.

Считаем, что число A содержится в числе B , если число A можно получить из числа B вычёркиванием некоторых цифр и перестановкой оставшихся.

Input

Входной файл состоит из не более, чем 60 неотрицательных чисел, меньших 10^{13} , записанных по одному в строке. Входной файл завершается строкой, содержащей число 0, обрабатывать которое не требуется.

Output

Для каждого числа выведите текст “Hidden squares in ”, затем — заданное число, затем — список чисел и их квадратов, построенных из заданного числа, отсортированный по возрастанию. Точный формат вывода указан в примере к задаче (знаки ‘*’ и ‘=’ обрамляются единичными пробелами). Ответы для разных чисел должны разделяться пустой строкой.

Example

standard input	standard output
1369	Hidden squares in 1369
27	1 * 1 = 1
10	3 * 3 = 9
0	6 * 6 = 36
	13 * 13 = 169
	19 * 19 = 361
	31 * 31 = 961
	Hidden squares in 27
	Hidden squares in 10
	0 * 0 = 0
	1 * 1 = 1

Problem E. Окружности

Input file: *standard input*
Output file: *standard output*
Time limit: 2 секунды
Memory limit: 256 мегабайт

Это интерактивная задача.

На плоскости находятся n ($1 \leq n \leq 2000$) окружностей с целочисленными координатами центров и целочисленными положительными радиусами, лежащие полностью внутри квадрата X с вершинами $(0, 0)$, $(20, 20)$. Стороны квадрата параллельны осям координат.

Имея возможность узнать количество окружностей, содержащих внутри себя или на границе данную точку, определите n .

Interaction Protocol

Чтобы сделать запрос, программа должна вывести два вещественных числа (разделённые единичным пробелом) — координаты точки, лежащей в квадрате X . В ответ она получит одно число — количество окружностей, содержащих данную точку.

При выводе запроса или ответа обязательно выводите перевод строки и выполняйте `flush()` в случае использования буферизованного вывода.

Если нужно вывести ответ, программа должна сообщить об этом и вывести ответ в виде целого числа.

Количество запросов не должно превышать 5 000.

Output

Программа должна выводить либо строку, содержащую два вещественных числа x и y ($0 \leq x, y \leq 20$) с точностью до 5 знаков после запятой (можно выводить числа с большей точностью, но программа жюри их округлит до 5 знаков) — координаты точки, либо слово «Done» и через пробел ответ на задачу.

Input

Входной поток содержит строки с целыми числами — ответы на запросы.

Examples

standard input	standard output
2	1.0 1.0
2	1.0 0.0
2	0.0 1.0
2	2.0 1.0
2	1.0 2.0
1	5.0 5.0
1	5.0 7.0
1	5.1 6.0
	Done 3

В примере две окружности имеют центр в $(1, 1)$ и радиус 1, и ещё одна имеет центр в $(5, 6)$ и радиус 2.

Problem F. Редактор

Input file: *standard input*
Output file: *standard output*
Time limit: 8 секунд
Memory limit: 256 мегабайт

В рамках программы импортозамещения вам поручено разработать один из модулей будущего отечественного текстового редактора. Модуль должен вычислять расстояние между двумя строками. Редактор умеет выполнять две операции:

1. Удалить K последовательных символов из строки X .
2. Вставить K последовательных символов в строку X .

Стоимость каждой операции равна квадратному корню из K , где K — количество вставленных или удаленных символов. Расстояние между строками X и Y — это минимальная стоимость операций, необходимых, чтобы превратить X в Y .

Input

Первая строка входного файла содержит натуральное число T — количество тестов, $T \leq 100$. Далее следуют тесты. Каждый тест в первой строке содержит натуральные числа N и M , $1 \leq N, M \leq 200$ — длины строк X и Y . Следующие две строки содержат X и Y , состоящие из заглавных букв латинского алфавита.

Output

Для каждого теста в отдельной строке необходимо вывести расстояние между X и Y с двумя знаками после десятичной точки.

Example

standard input	standard output
1 4 4 ACDD ADGD	2.00

Note

Чтобы превратить строку “ACDD” в “ADGD”, необходимо выполнить две операции по одному символу, “ACDD \rightarrow ADD \rightarrow ADGD”, а именно: удаление символа ‘C’ и вставку символа ‘G’.

Problem G. Каникулы

Input file: *standard input*
Output file: *standard output*
Time limit: 4 секунды
Memory limit: 256 мегабайт

Вася решил спланировать летние каникулы заранее. Он собирается во время каникул поехать на море... с другой стороны, в городе, в котором живёт Вася, летом планируются концерты рок-групп, а также футбольные матчи. Все эти мероприятия Васе посетить тоже хочется. Поэтому, вооружившись прогнозом погоды, расписанием чемпионата по футболу и графиком турне рок-групп, Вася для каждого дня определил «коэффициент отдыха» — целое число, большее нуля в случае, если этот день лучше провести на море, и меньшее нуля в случае, если этот день лучше провести в городе.

С морем город, в котором живёт Вася, соединяет регулярный авиарейс, выполняемый по ночам. Авиакомпания, обслуживающая этот рейс, проводит предновогоднюю акцию: продаёт k билетов «туда-обратно» на различные рейсы в летний период по цене одного.

Вася хочет распланировать свой отдых таким образом, чтобы сумма «коэффициентов отдыха» за все дни, которые он проведёт на море, была максимальной, а количество полётов на море и обратно не превышало k . Ваша задача — вычислить соответствующую максимальную сумму.

Input

Первая строка входного файла содержит два целых числа n и k ($1 \leq k \leq n \leq 10^6$) — продолжительность каникул в днях и количество билетов, продаваемых по акции, соответственно. Во второй строке заданы n целых чисел, каждое из которых не превосходит по абсолютной величине 10^9 . i -е из этих чисел задаёт «коэффициент отдыха» i -го дня каникул.

Output

Выведите одно число — максимальный суммарный «коэффициент отдыха» за те дни, которые Вася проведёт на море.

Example

standard input	standard output
5 2 7 -3 4 -9 5	13

Problem H. Авиалинии

Input file: *standard input*
Output file: *standard output*
Time limit: 1 секунда
Memory limit: 256 мегабайт

В некотором государстве есть M аэропортов и K авиакомпаний. Авиакомпаниями организовано N прямых рейсов между аэропортами. Каждый рейс является двусторонним (то есть состоит из прямого и обратного) и обслуживается одной авиакомпанией. Между двумя аэропортами может быть организовано несколько рейсов, если никакие два из этих рейсов не принадлежат ни одной авиакомпании. При этом для каждой пары аэропортов существует маршрут между этими аэропортами (прямой или с некоторым количеством пересадок).

Инспектору Министерства Транспорта поручено проверить качество обслуживания пассажиров. Для того, чтобы выполнить поручение, инспектор должен по одному разу воспользоваться каждым из N рейсов (в одну сторону). При этом во избежание обвинений в предвзятости инспектор должен всякий раз пересаживаться на самолёт другой авиакомпании (то есть он не может выбрать для двух последовательных сегментов пути рейсы, обслуживаемые одной и той же авиакомпанией). При этом первый и последний полёты также должны обслуживаться различными авиакомпаниями.

С целью оптимизации бюджетных расходов министерство хочет составить маршрут для инспектора таким образом, что:

- Маршрут начинается в столице (аэропорт с номером 1);
- каждый последующий полёт начинается в том же аэропорте, в котором закончился предыдущий;
- маршрут заканчивается в столице.

Ваша задача — найти такой маршрут или установить, что пожелания министерства невыполнимы.

Input

В первой строке входного файла заданы три целых числа M , N и K ($2 \leq M \leq 1000$; $1 \leq N \leq 10^4$; $1 \leq K \leq 10$). Каждая из последующих N строк описывает один рейс и содержит три целых числа: номера аэропортов, которые соединяет рейс, и номер авиакомпании, этот рейс обслуживающей. Аэропорты занумерованы различными целыми числами от 1 до M , а авиакомпании — от 1 до K . Рейсы занумерованы от 1 до N в порядке их перечисления во входном файле.

Output

Если пожелания министерства невыполнимы, выведите в первой строке слово “No”. В противном случае первая строка должна содержать слово “Yes”, а каждая из последующих N строк должна содержать номера рейсов в порядке, в котором инспектор должен ими воспользоваться. В случае, если существует несколько решений, выведите любое из них.

Example

standard input	standard output
4 7 3 3 2 1 1 2 2 3 4 1 3 4 2 3 4 3 1 2 3 2 4 2	Yes 2 1 5 4 3 7 6
4 7 2 3 2 1 1 2 1 3 4 1 3 1 2 3 1 1 4 2 1 4 2 2	No