Задача A. Конспирация

Имя входного файла: `conspiracy.in`
Имя выходного файла: `conspiracy.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Петя и Вася играют в шпионов. Главная цель этой игры — обмениваться секретными сообщениями, сохраняя конспирацию. Для этого Петя и Вася используют особый шифр.

Метод шифровки и дешифровки очень прост: каждой букве α русского алфавита сопоставляется некоторая последовательность букв латинского алфавита p_α. Затем в сообщении все вхождения каждой буквы α заменяются на соответствующее ей p_α. А чтобы совсем запутать вероятного противника, Петя и Вася договорились, что будут отправлять n слов, значимым из которых будет только лексикографически минимальное слово. Напомним, что строка a лексикографически меньше строки b, если существует такое i, что для любого $j < i$ выполняется равенство $a_j = b_j$ и при этом $a_i < b_i$, или строка a является префиксом строки b.

Недавно Петя предложил Васе использовать следующую таблицу преобразования букв:

а	a	з	z	п	p	ч	tch
б	b	и	i	р	r	ш	sch
в	v	й	j	с	s	щ	scsh
г	g	к	k	т	t	ы	yy
д	d	л	l	у	u	э	ye
е	e	м	m	ф	f	ю	yu
ё	yo	н	n	х	kh	я	ya
ж	zh	о	o	ц	tc		

Однако теперь Вася не успевает дешифровывать сообщения Пети вовремя и просит вас помочь ему — определить номер слова, которое до шифрования, записанное русскими буквами, было лексикографически минимальным.

Формат входного файла

Первая строка входного файла содержит одно целое число n $(1 \le n \le 1000)$ — количество слов в шифровке. Следующие n строк содержат по одному слову, состоящему из маленьких латинских букв — текст шифровки, которую получил Вася. Длина слова не превышает 50.

Формат выходного файла

В выходной файл выведите номер слова шифровки, которое до шифрования было лексикографически минимальным в русском алфавите. Шифровки нумеруются с единицы в порядке появления во входном файле.

Примеры

conspiracy.in	conspiracy.out
6 yolka jeti yabloko segodnya schtab kaktus	1

Примечание

Пояснение к примеру: исходное сообщение состояло из слов «`ёлка`», «`йети`», «`яблоко`», «`сегодня`», «`штаб`», «`кактус`».

Задача B. Дома

Имя входного файла:	`houses.in`
Имя выходного файла:	`houses.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

В Стране хотят построить город программистов Осколково. Город по плану будет большой. В нём будет N улиц, на каждой из которой по a_i домов. Правительство решило применить инновации, и вместо обычных табличек с номерами домов они хотят повесить таблички с номерами домов, записанными в двоичном представлении.

После утверждения такой нумерации в городе Осколково, был произведён тендер на изготавливает табличек с цифрами 0 и 1. Выигравшая компания изготовляет табличку с цифрой 0 за a рублей, а табличку с цифрой 1 за b рублей. Теперь правительство хочет узнать, сколько будет стоит изготовление табличек на дома.

Формат входного файла

Первая строка входного файла содержит три натуральных числа N ($1 \leq N \leq 100$) и a, b ($1 \leq a, b \leq 100$) — количество улиц в городе и стоимости изготовления цифр 0 и 1.

В следующих N строках содержится натуральное число a_i ($1 \leq a_i \leq 100$) — количество домов на i-й улице и a_i натуральных чисел c_j ($1 \leq c_j \leq 100$) — номера домов на этой улице, для которых будут изготовлены таблички. Для одного дома может потребоваться изготовить более одной таблички.

Формат выходного файла

Вывести одно натуральное число — стоимость изготовления табличек.

Примеры

houses.in	houses.out
1 1 2 3 2 2 3	10

Задача C. k-сортировка

Имя входного файла:	`ksort.in`
Имя выходного файла:	`ksort.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

В этом году Гриша поступил в Университет ИТ. В Университете ИТ очень много новых предметов, интересных и не очень. Особенно Грише нравится предмет «Алгоритмы и структуры данных». На последней лекции были рассказаны алгоритмы сортировки. Гриша — очень амбициозный молодой человек и хочет изобрести свой алгоритм, который впоследствии будет назван именем его любимого дедушки. Вдохновившись чтением многотомника Кнута, Гриша решил модернизировать какой-нибудь уже существующий алгоритм сортировки натуральных чисел, наложив следующее ограничение. Любые два элемента можно менять местами, только если они сравнимы по модулю некоторого натурального числа k, то есть дают одинаковые остатки при делении на k. Но все инновационные методы требуют проверки, поэтому Гриша обратился за помощью к Вам!

Проверьте, сможет ли новая версия алгоритма отсортировать заданный массив натуральных чисел.

Формат входного файла

Первая строка входного файла содержит два числа n ($1 \le n \le 1000$) и k ($1 \le k \le 10^9$) — количество элементов в массиве и число, по модулю которого сравниваются элементы массива.

Вторая строка входного файла содержит n целых чисел a_i — элементы массива ($1 \le a_i \le 10^9$).

Формат выходного файла

В выходной файл выведите «YES», если алгоритм сможет отсортировать заданный массив и «NO» — в обратном случае.

Примеры

ksort.in	ksort.out
5 2 5 4 3 2 1	YES
3 2 2 3 1	NO

Задача D. Огни светофора

Имя входного файла:	`lights.in`
Имя выходного файла:	`lights.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Боб — известный ученый. Так как он живет недалеко от места своей работы, он часто ходит пешком. По пути из дома на работу он встречает один светофор. Светофор состоит из лампочек трех цветов: красного, желтого и зеленого.

Светофор работает по описанным ниже правилам.

1. В течение g секунд светит зеленый свет
2. В течение gb секунд мигает зеленый, при этом ровно половину времени лампочка светится
3. В течение y секунд светит желтый
4. В течение r секунд светит красный
5. В течение ry секунд светят одновременно красный и желтый

После пятого пункта снова выполняется первый. Переключение между режимами происходит мгновенно.

Боб хочет узнать для каждой лампочки, сколько времени она светится в течении дня, если известно, что день длится T секунд. При этом Боб считает, что в самом начале дня начал светиться зеленый. В режиме, когда мигает зеленый свет, считается что с его начала до любого момента времени лампочка светится ровно половину времени.

Формат входного файла

Первая строка входного файла содержит пять целых чисел g, gb, y, r, ry ($1 \le g, gb, y, r, ry \le 100$), gb — четно.

Вторая строка содержит целое число T ($1 \le T \le 10000$) — продолжительность дня в секундах.

Формат выходного файла

В выходной файл выведите три целых числа: сколько времени горела красная, желтая и зеленая лампочка, соответственно, за один день.

Примеры

lights.in	lights.out
`2 2 1 2 1` `6`	`1 1 3`

Задача E. Зарплата

Имя входного файла:	`salary.in`
Имя выходного файла:	`salary.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Дима совсем недавно устроился на работу. Так как он очень ценный работник, его зарплата будет расти не по дням, а по часам. Если точнее, то каждый месяц количество рублей, составляющих его зарплату, будет равно возведенному в третью степень номеру месяца. Так, в первый месяц он получит 1 рубль, во второй — 8, в третий — 27 и так далее.

Как всем известно, законы о налогообложении — вещь сложная. Недавно в стране, в которой работает Дима, был принят новый закон. Суть его заключается в том, что процент налога, ежемесячно выплачиваемый сотрудником вычисляется по следующей формуле:

- Записывается зарплата сотрудника за первый месяц, к ней справа приписывается его зарплата за второй месяц и так далее, пока суммарное количество цифр в записанном числе не будет больше k
- k-ая цифра записанного числа и является искомым процентом

Для каждой организации было введено свое k. Дима знает это число для организации, в которой он работает. И теперь он хочет узнать, сколько процентов заработка он будет ежемесячно отдавать в качестве налога.

Формат входного файла

Первая строка входного файла содержит единственное целое число k ($1 \le k \le 10^7$).

Формат выходного файла

Выведите в выходной файл одно целое число x ($0 \le x \le 9$) — искомый процент.

Примеры

salary.in	salary.out
3	2
7	1

Примечание

При вычислении налога во втором примере получается следующая строка: 182764125. Седьмая цифра при этом равна 1.

Задача F. Траволатор

Имя входного файла:	`travelator.in`
Имя выходного файла:	`travelator.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Боб — известный ученый, и поэтому он часто посещает различные конференции. Во время очередной поездки Боб опаздывал на самолет, и поэтому ему необходимо было попасть в терминал A как можно быстрее. Известно, что в аэропорту есть горизонтальный траволатор, благодаря которому Боб может передвигаться по аэропорту быстрее. Бобу известен план аэропорта, с обозначенными на нем входом, терминалом A и концами траволатора.

Если аэропорт представить в виде координатной плоскости, то вход в аэропорт находится в точке (0,0), терминал A в точке (x_1,y_1), начало и конец траволатора в точках (x_2,y_2) и (x_3,y_3). Также известно, что скорость Боба v, скорость траволатора w и по траволатору можно двигаться только из начала в конец. Помогите Бобу найти минимальное время, за которое он доберется от входа в аэропорт к терминалу A. При этом, так как у эскалатора есть перила, запрещено его пересекать, даже двигаясь вне траволатора, а можно только войти на него в точке начала траволатора и двигаться со скоростью w вплоть до конца траволатора.

Формат входного файла

Первая строка входного файла содержит шесть целых чисел x_1, y_1, x_2, y_2, x_3, y_3 $(-10000 \le x_1, x_2, x_3 \le 10000, 0 < y_1, y_2, y_3 \le 10000)$ — координаты терминала A, начала и конца траволатора. Вторая строка содержит два целых числа v, w $(0 < v \le 50, 0 < w \le 50)$ — скорости Боба и траволатора.

Известно что терминал A и вход в аэропорт не лежат на траволаторе.

Формат выходного файла

Выведите минимальное время, за которое Боб может добраться от входа до терминала A. Ответ считается правильным, если он отличается от минимального не более чем на 10^{-5}.

Примеры

travelator.in	travelator.out
0 3 0 1 0 2 1 4	2.25
0 3 -2 1 2 1 1 4	5.064495

Примечание

В первом примере Бобу выгодно воспользоваться траволатором.

Задача G. Треугольник

Имя входного файла:	`triangle.in`
Имя выходного файла:	`triangle.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Отдыхая на даче у Петиной бабушки, Петя и Вася решили построить секретный штаб для своей шпионской организации. Для размещения всей шпионской техники им необходимо помещение площадью ровно S квадратных метров. Стены штаба приходится строить из подручного материала, найденного у бабушки в сарае. Этим материалом оказались деревянные доски, каждая из которых имела длину ровно один метр. Выбирая форму штаба, Вася вспомнил последний урок геометрии, на котором они проходили прямоугольные треугольники, и предложил построить штаб в форме этой фигуры. Тем более, тогда они легко смогут посчитать его площадь и убедиться, что она равна требуемой.

Но тут оказалось, что пилы нет и поэтому доски никак не распилить. Васе с Петей придется строить штаб в форме прямоугольного треугольника, все стороны которого выражаются целым числом метров. Им стало интересно, сколько таких треугольников существует. При этом треугольники, имеющие одинаковые длины сторон, но отличающиеся их порядком, считаются одинаковыми.

Формат входного файла

Входной файл содержит единственное целое число S ($1 \le S \le 10\,000$) — требуемая площадь штаба.

Формат выходного файла

В выходной файл выведите одно целое число — количество возможных планов постройки штаба, площадь которого S квадратных метров.

Примеры

triangle.in	triangle.out
1	0
6	1

Задача H. Где я?

Имя входного файла:	`where.in`
Имя выходного файла:	`where.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Дедушка Марат живет в далеком-далеком городе Ч. Дедушка очень любит ходить в гости, иногда он уходит на несколько дней, обходя при этом очень-очень много своих друзей. Дедушка, уходя из очередного дома, всегда идет только к друзьям хозяев этого дома. К некоторым Дедушка Марат мог заходить по нескольку раз. Дедушка мог даже заходить к себе домой попить чаю с внуками. Однако Дедушка очень забывчив, поэтому он иногда попросту забывает вернуться домой. Его внуки очень волнуются за него, поэтому всегда находят его и возвращают его домой. За несколько лет внуки поняли, что прежде чем они успевают найти Дедушку Марата, он успевает обойти ровно k друзей (внуки тоже считаются друзьями).

Несколько дней назад Дедушка Марат снова ушел погостить, и внуков интересует, где же они могут его встретить? Помогите им узнать ответ на этот вопрос.

Формат входного файла

Первая строка входного файла содержит три числа n, m и k, где n — количество домов в городе Ч., а m — количество пар друзей ($1 \le n \le 1\,000$, $1 \le m \le 200\,000$, $1 \le k \le 10^9$).

Следующие m строк содержат описания пар друзей, по одному на каждой строке. Описание состоит из двух чисел — номера домов, хозяева которых дружат (если хозяева дома i дружат с хозяевами дома j, то и хозяева дома j дружат с хозяевами дома i

Дедушка Марат и внуки живут в доме с номером 1.

Формат выходного файла

В первой строке выходного файла должно быть число p — количество домов, в которых мог оказаться Дедушка Марат. Во второй строке должно быть p чисел — номера домов, в которых мог оказаться Дедушка Марат, в возрастающем порядке.

Примеры

where.in	where.out
3 3 3 1 2 1 3 2 3	3 1 2 3