

Задача А. Арифметическая прогрессия

Имя входного файла: `a.in`
Имя выходного файла: `a.out`
Ограничение по времени: 1 с
Ограничение по памяти: 256 Мб

Требуется найти арифметическую прогрессию из натуральных чисел a_1, a_2, \dots, a_n , с разностью d , обладающую свойством: $a_k^2 + 1$ — простое при всех $k = 1, 2, \dots, n$. Среди всех таких прогрессий следует выбрать ту, которая состоит из максимального числа элементов.

Разность прогрессии d означает, что для всех $k = 2, 3, \dots, n$ выполняется $a_k - a_{k-1} = d$.

Формат входного файла

Входной файл содержит несколько тестов. В каждой строке записано целое число d — разность прогрессии ($1 \leq d \leq 9999$). В десятичной записи числа d цифра 0 не встречается. Все числа во входном файле различны.

Формат выходного файла

Для каждого теста выведите в выходной файл по одной строке, содержащей два числа. Первое число — максимальная длина арифметической прогрессии. Второе число — её первый элемент. Среди всех прогрессий максимальной длины выберите прогрессию с наименьшим первым элементом.

Пример

<code>a.in</code>	<code>a.out</code>
6	3 4
311	1 1

Задача В. Мирные кони

Имя входного файла: `b.in`
 Имя выходного файла: `b.out`
 Ограничение по времени: 3 с
 Ограничение по памяти: 256 Мб

Рассмотрим следующую задачу. Есть стандартная шахматная доска размером 8 на 8 клеток. Необходимо расставить на ней N коней, так, чтобы они не били друг друга, при условии, что на i -ой горизонтали должно стоять ровно r_i коней, а на j -ой вертикали должно стоять ровно c_j коней, где сумма всех r_i равна сумме всех c_j и равна N . Шахматный конь — это фигура, ход которой состоит в перемещении её на две клетки в горизонтальном или вертикальном направлении, а затем ещё на одну — в перпендикулярном направлении. Считается, что под боем находятся только те клетки, в которых конь может закончить свой ход. В каждой клетке может стоять не более одного коня.

Требуется по заданным значениям r_i найти такие значения для c_j , чтобы исходная задача имела ровно одно решение.

Формат входного файла

В первой строке входного файла записаны восемь чисел — r_1, r_2, \dots, r_8 ($0 \leq r_i \leq 8$).

Формат выходного файла

В выходной файл выведите через пробел искомые восемь чисел — c_1, c_2, \dots, c_8 ($0 \leq c_j \leq 8$). Если невозможно подобрать требуемым образом эти числа, то выведите через пробел восемь чисел -1 . Если подходящих ответов несколько, то выведите из них наименьший лексикографически (т.е. с минимально возможным значением c_1 , если таких несколько, то с минимально возможным значением c_2 и т.д.).

Примеры

<code>b.in</code>	<code>b.out</code>
3 3 3 3 3 3 3 3	0 8 0 0 8 0 0 8
4 4 4 4 4 4 4 4	-1 -1 -1 -1 -1 -1 -1 -1
2 3 4 3 2 3 4 3	0 2 4 2 4 4 4 4

Задача С. Четыре точки

Имя входного файла: `c.in`
Имя выходного файла: `c.out`
Ограничение по времени: 1 с
Ограничение по памяти: 256 Мб

На плоскости заданы четыре точки. Требуется построить окружность, равноудалённую от всех этих точек.

Расстояние от точки A до окружности — это минимальное из расстояний от точки A до какой-либо точки B , лежащей на этой окружности.

Формат входного файла

В первой строке входного файла содержится одно число T — количество тестов ($1 \leq T \leq 10^4$). В каждой из следующих T строк записаны координаты четырёх точек $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4$ — целые числа, не превосходящие 100 по абсолютному значению.

Формат выходного файла

Для каждого теста выведите в отдельной строке число K — количество различных окружностей, удовлетворяющих условию. Далее в K строках выведите по три числа — координаты центра окружности и её радиус (x_i, y_i, r_i) . Если возможно построить более 42 различных окружностей, выведите вместо числа K строку “-1”. Сами окружности в этом случае выводить не следует.

Допускается абсолютная погрешность не более 10^{-5} .

Пример

<code>c.in</code>	<code>c.out</code>
3	-1
0 0 0 0 0 0 0 0	5
4 0 4 16 0 8 8 8	4 11 8
0 0 1 1 0 1 1 0	4 5 8
	-2 8 6
	10 8 6
	4 8 6
	-1

Задача D. Римские числа

Имя входного файла: d.in
Имя выходного файла: d.out
Ограничение по времени: 1 с
Ограничение по памяти: 256 Мб

Дана строка из символов 'I', 'V', 'X', 'L', 'C', 'D' и 'M'. Требуется разбить её на несколько строк так, чтобы каждая получившаяся строка представляла собой корректное римское число, а сумма этих чисел была бы минимально возможной.

Корректным римским числом будем называть число, которое получается по следующим правилам. Сначала выписывается символ 'M' столько раз, сколько целых тысяч содержится в числе. Например, в числе 2045 содержится две целых тысячи, и его запись начнётся с символов "MM". После этого берём остаток от деления его на 1000. В зависимости от того, сколько целых сотен содержит результат, добавляем к римской записи следующую строку: "C" — для одной сотни, "CC" — для двух, "CC", "CCC", "CD", "D", "DC", "DCC", "DCCC", "CM" — для трёх, четырёх, ..., для девяти сотен соответственно. Остаток от деления числа 2045 на 1000 равен 45. Целых сотен в нём нет, поэтому к римской записи ничего не добавляем.

Далее берём остаток от деления числа на 100 и, в зависимости от количества целых десятков в результате, по тем же правилам добавляем к римской записи "X", "XX", "XXX", "XL", "L", "LX", "LXX", "LXXX" или "XC" (либо ничего не добавляем, если целых десятков нет). Соответственно, для числа 2045 мы должны добавить к римской записи строку "XL".

После этого берётся остаток от деления на 10 исходного числа, и, в зависимости от результата, добавляется "I", "II", "III", "IV", "V", "VI", "VII", "VIII", "IX" — для 1, 2, ..., 9 соответственно, либо, в случае нулевого остатка, ничего не добавляется. Для числа 2045 по этому правилу мы должны прибавить строку "V".

В итоге для числа 2045 получаем римскую запись "MMXLV".

Также на римские числа вводится дополнительное ограничение, что никакой символ в их записи не может встречаться более трёх раз подряд. То есть будем считать, что корректная римская запись существует только у чисел от 1 до 3999 включительно.

Формат входного файла

Во входном файле содержится одна непустая строка длиной не более 20000 символов, состоящая из заглавных латинских букв, указанных в условии ('I', 'V', 'X', 'L', 'C', 'D' и 'M').

Формат выходного файла

В выходной файл выведите одно число — минимально возможную сумму чисел.

Примеры

d.in	d.out
IVI	5
IIIIIV	7
CDILMVX	1466
M	1000
IX	9

Задача Е. Камни

Имя входного файла: `e.in`
Имя выходного файла: `e.out`
Ограничение по времени: 2 с
Ограничение по памяти: 256 Мб

Двое играют в игру. Изначально есть одна кучка из N камней. За один ход разрешается взять любую кучку и разбить её на несколько так, чтобы количества камней в новых кучках отличались бы не менее, чем на K . Например, при $K = 2$ кучку из 9 камней можно разбить на кучки (1, 8), (2, 7), (3, 6) и (1, 3, 5). Тот, кто не сможет сделать очередной ход, проигрывает. При заданном N определите все K ($0 \leq K \leq N$), при которых выигрывает второй игрок.

Формат входного файла

В первой строке входного файла записано натуральное число N ($1 \leq N < 100$).

Формат выходного файла

В первую строку выходного файла выведите общее количество искомых чисел, а во вторую — сами числа в порядке возрастания.

Пример

<code>e.in</code>	<code>e.out</code>
9	3 6 8 9

Задача F. Две строки

Имя входного файла: `f.in`
 Имя выходного файла: `f.out`
 Ограничение по времени: 1 с
 Ограничение по памяти: 256 Мб

Даны две строки S_1 и S_2 , состоящие из строчных латинских букв. В каждой из этих строк выбирается некоторая начальная позиция (k_1 и k_2). Затем, начиная с этой позиции, по порядку выписываются все символы строки. После выписывания последнего символа переходят к первому и продолжают выписывать подряд все символы. Например, если строка S_1 равна "CAB", а $k_1 = 2$ (нумерация начинается с единицы), то получаем строку "ABCABСABСABСABС...". Если строка S_2 равна "BCACAC", а $k_2 = 3$, то получаем строку "ACACBCACACBCACAC...". Так получаем две бесконечные последовательности из символов. Обозначим их T_1 и T_2 .

Требуется для заданных строк S_1 и S_2 найти такие k_1 и k_2 , чтобы значение выражения $\lim_{n \rightarrow \infty} \frac{\sum_{i=1}^n eq(T_1[i], T_2[i])}{n}$ было максимально возможным (здесь $T_1[i]$ — i -ый символ строки T_1 , $eq(a, b)$ равно 1, если символы a и b совпадают, или 0, если a и b различаются). То есть нужно максимизировать среднее количество совпавших символов в полученных последовательностях.

Формат входного файла

В первой строке входного файла записана непустая строка S_1 . Во второй строке входного файла записана непустая строка S_2 . Длина каждой строки не превосходит 2000 символов.

Формат выходного файла

В выходной файл выведите искомые числа k_1 и k_2 ($1 \leq k_1 \leq \text{length}(S_1)$, $1 \leq k_2 \leq \text{length}(S_2)$).

Примеры

f.in	f.out
CAB BCACAC	1 2
ABRACADABRA TEST	1 1
AAAB BABA	1 2