

Задача А. Плавные числа

Имя входного файла: numbers.in
Имя выходного файла: numbers.out
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Назовём натуральное число плавным, если разность любых двух его соседних цифр не превосходит по модулю единицы. Вам необходимо определить количество N -значных плавных чисел.

Формат входных данных

В единственной строке входного файла одно число N ($1 \leq N \leq 20$).

Формат выходных данных

Вывести одно число — искомое количество плавных чисел.

Примеры

numbers.in	numbers.out
2	26

Задача В. Наибольшая последовательнократная подпоследовательность

Имя входного файла: sequence.in
Имя выходного файла: sequence.out
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Для заданной числовой последовательности a_1, a_2, \dots, a_n требуется найти длину максимальной последовательнократной подпоследовательности.

Для последовательнократной подпоследовательности $a_{k_1}, a_{k_2}, \dots, a_{k_t}$ ($k_1 < k_2 < \dots < k_t$) верно, что $a_{k_i} | a_{k_j}$ при $1 \leq i < j \leq t$ (утверждение « $a|b$ » эквивалентно « b кратно a »). Подпоследовательность из одного элемента полагается последовательнократной по определению.

Формат входных данных

В первой строке входного файла записано одно натуральное число N ($1 \leq N \leq 1000$) — количество чисел в исходной последовательности. Далее следует N натуральных чисел, не превосходящих $2 \cdot 10^9$ — сама последовательность.

Формат выходных данных

Вывести единственное число, равное искомому количеству.

Примеры

sequence.in	sequence.out
4 3 6 5 12	3

Задача С. Банкомат

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

В некотором государстве в обращении находятся банкноты определенных номиналов. Национальный банк хочет, чтобы банкомат выдавал любую запрошенную сумму при помощи минимального числа банкнот, считая, что запас банкнот каждого номинала неограничен. Помогите Национальному банку решить эту задачу.

Формат входных данных

Первая строка входных данных содержит натуральное число N не превосходящее 100 — количество номиналов банкнот в обращении. Вторая строка входных данных содержит N различных натуральных чисел x_1, x_2, \dots, x_N , не превосходящих 10^6 — номиналы банкнот. Третья строка содержит натуральное число S , не превосходящее 10^6 — сумму, которую необходимо выдать.

Формат выходных данных

В первую строку выходного файла выведите минимальное число слагаемых (или -1, если такого представления не существует). Во вторую строку выведите это представление в любом порядке.

Примеры

atm.in	atm.out
5 1 3 7 12 32 40	3 32 7 1

Задача D. Получите в сумме ноль

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

Даны целые неотрицательные числа a_1, a_2, \dots, a_n . Определите, какой знак следует поставить перед каждым из них, чтобы результат сложения данных чисел с учетом выбранных знаков равнялся нулю. Гарантируется, что способ расставить знаки таким образом существует.

Формат входных данных

В первой строке записано целое число n ($1 \leq n \leq 100$). Во второй строке через пробел записаны числа a_i ($0 \leq a_i \leq 500$).

Формат выходных данных

Выведите единственную строку из n символов. i -й символ строки должен соответствовать знаку, который требуется поставить в сумме перед i -м числом, чтобы получить в

результате ноль.

Примеры

zerosum.in	zerosum.out
4	---+
2 3 0 1	

Задача Е. Строка

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

Дед Мороз составил важное сообщение, которое нужно передать другим Дедам Морозам. На всякий случай Дед Мороз записал свое сообщение два раза.

Потом он ненадолго отвлекся, а в это время один из эльфов прибежал в дом и решил пошалить. Он взял и приписал к сообщениям спереди и сзади какую-то абракадабру.

Теперь Деду Морозу надо найти наибольшую подстроку, которая могла бы быть его сообщением. Помогите ему.

Например, пусть сообщение Деда Мороза имело вид `'xzz'`. После этого эльф приписал к первому сообщению спереди букву `a`, сзади — ничего; ко второму — спереди ничего, сзади `b`. В итоге получились строки `'axxz'` и `'xxzb'`.

Формат входных данных

Входной файл содержит две строки, состоящие из строчных латинских букв, длиной не более 5000. Гарантируется, что обе строки во входном файле непустые.

Формат выходных данных

Выведите строку — предполагаемое сообщение.

Примеры

lcstr.in	lcstr.out
caa	aa
aab	

Задача F. Кино

Имя входного файла: `cinema.in`
Имя выходного файла: `cinema.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Трое друзей летом посмотрели несколько фильмов. Для каждого школьника известно, какие фильмы и в каком порядке он посмотрел (естественно, если фильм кому-то из ребят сильно понравился, он мог его пересмотреть несколько раз). Так как друзья любят смотреть фильмы вместе, втроем они посмотрели максимальное возможное количество. Сколько же раз они встречались вместе?

Формат входных данных

В первой строке входного файла заданы три числа: N , M и K — количества фильмов, просмотренных каждым из друзей ($1 \leq N, M, K \leq 300$). В следующих трех строках выписаны номера фильмов, просмотренных ими. У друзей огромные планы на будущее, поэтому фильмы нумеруются числами между 1 и 10^9 .

Формат выходных данных

В первой строке выведите единственное число — максимальное кол-во просмотренных фильмов. В следующей строке выведите через пробел номера просмотренных фильмов.

Примеры

cinema.in	cinema.out
3 3 4	2
1 2 3	1 3
1 3 10	
3 1 10 3	