

Задача А. ВСОШ по Информатике

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

НИЯУ МИФИ в 2026 году впервые был организатором ВСОШ по информатике.

Примечание автора задачи.
Имя реального организатора в задаче изменено, совпадение с реальностью случайно.

Две недели назад Володя занимался установкой очень большого количества ноутбуков участников для ВСОШ по информатике. Но возникла серьёзная проблема: дедлайн по установке оставшихся ноутбуков через два часа, а процесс распаковки, подключения к питанию/интернету и установки операционной системы со всем необходимым ПО для участия занимает много времени по отдельности. Володя не растерялся и вспомнил, что до этого закупил N удлинителей, в каждом из которых c_i разъёмов, а также что подключённые ноутбуки можно настраивать удалённо по сети.

С учётом того, что розетка в помещении всего одна, помогите Володе понять, как ему соединить все удлинители и подключить их к питанию так, чтобы число ноутбуков, которое должно быть подготовлено к олимпиаде, было максимальным.

Считайте, что объём нагрузки на электросеть позволяет пользоваться подключением удлинителей друг к другу и на безопасность это никак не влияет.

Формат входных данных

В первой строке вводится число N ($1 \leq N \leq 10^5$) — количество удлинителей, которые можно задействовать.

Во второй строке через пробел вводятся N чисел c_i — количество разъёмов в i -м удлинителе соответственно.

Формат выходных данных

Выведите ответ на задачу.

Система оценки

Каждый тест данной задачи кроме примеров из условия оценивается в $\frac{10}{13}$ балла с округлением до 2-х знаков после запятой по математическим правилам, при этом при неверном ответе на произвольный тест тестирование задачи продолжается до конца.

Примеры

laptop.in	laptop.out
1 3	3
3 3 3 3	7

Задача В. Отсортировать заказы

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

В крупной логистической системе каждому заказу присваивается уникальный числовой идентификатор от 1 до n .

Для автоматизации обработки заказов используются специальные правила маршрутизации: заказ может быть обработан определённым узлом системы только в том случае, если его идентификатор делится на заданное число k .

Система перебирает идентификаторы в *лексикографическом порядке* — то есть так, как если бы числа были записаны в виде строк и отсортированы по алфавиту (например, 1, 10, 100, 101, ..., 11, 110, ..., 2, 20, ...).

Для каждого запроса требуется определить, какой первый подходящий идентификатор в таком порядке будет обработан системой.

Формат входных данных

Каждый тест содержит несколько пар чисел для дальнейшей обработки.

Каждая пара вводится через пробел в отдельной строке и содержит два целых числа n и k ($1 \leq n \leq 10^{18}$, $1 \leq k \leq n$).

Последняя строка входных данных содержит два нуля и обозначает конец ввода. Обрабатывать эти нули не требуется.

Формат выходных данных

Для каждой пары выведите одно целое число — наименьший в лексикографическом порядке идентификатор, не превышающий n , который делится на k .

Система оценки

Каждый тест данной задачи, кроме тестов из условия, оценивается в $\frac{5}{6}$ балла, с округлением до 2 знаков после запятой по математическим правилам, при этом в случае неверного ответа на произвольный тест тестирование решения продолжается до конца.

Пример

<code>ids.in</code>	<code>ids.out</code>
2000 17	1003
2000 20	100
2000 22	1012
0 0	

Задача С. Суммы — это просто!

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

На уроке математики проводится эксперимент с наборами чисел. Рассматриваются числа от 1 до N , которые необходимо распределить по нескольким группам.

Каждая группа считается *корректной*, если сумма всех чисел в ней является простым числом (то есть числом, имеющим ровно два натуральных делителя).

Требуется разбить числа на минимально возможное количество *корректных* групп.

Формат входных данных

В единственной строке вводится одно целое число N ($2 \leq N \leq 5\,000$).

Формат выходных данных

Выведите N целых чисел, разделённых пробелами, где i -е число должно обозначать номер группы, в которую помещено число i .

Группы нумеруются натуральными числами, начиная с 1.

Допускается вывод любого корректного разбиения, использующего минимальное количество групп.

Система оценки

Каждый тест данной задачи кроме тестов из условия оценивается в $\frac{5}{14}$ баллов с округлением до 2-х знаков после запятой по математическим правилам, при этом в случае неверного ответа на произвольный тест тестирование задачи продолжается до конца.

Пример

<code>primes.in</code>	<code>primes.out</code>
8	2 2 2 2 1 2 2 2

Замечание

Например, при $N = 8$ допустимое разбиение может выглядеть так: $\{1, 4, 5, 6, 7\}$ и $\{2, 3, 8\}$, так как суммы чисел в обеих группах являются простыми.

Задача D. Преобразуйте массив

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

В системе обработки данных имеются два массива одинаковой длины. Первый массив содержит n чисел a_i , второй — n чисел b_i .

Разрешается выполнять следующую операцию: за одну единицу времени можно поменять местами два соседних элемента первого массива (позиции i и $i + 1$).

Требуется преобразовать первый массив так, чтобы он полностью совпадал со вторым массивом.

Определите минимальное количество операций, необходимое для этого преобразования.

Формат входных данных

В первой строке задано целое число n ($1 \leq n \leq 100\,000$).

Во второй строке заданы n целых чисел $a_1, a_2, \dots, a_n$ — элементы первого массива.

В третьей строке заданы n целых чисел $b_1, b_2, \dots, b_n$ — элементы второго массива.

Все числа по модулю не превышают 10^6 .

Формат выходных данных

Выведите одно целое число — минимальное количество операций, необходимое для преобразования первого массива во второй.

Если это невозможно, выведите -1 .

Система оценки

Каждый тест данной задачи оценивается в $\frac{10}{17}$ балла с округлением до 2 знаков после запятой по математическим правилам, при этом в случае неверного ответа на произвольный тест тестирование задачи останавливается на этом тесте.

Примеры

<code>arrays.in</code>	<code>arrays.out</code>
5 3 2 3 4 5 5 4 3 2 3	7
8 1 2 1 2 1 2 1 2 2 1 2 1 2 1 2 2	-1

Задача Е. Таблица

Имя входного файла:	<code>table.in</code>
Имя выходного файла:	<code>table.out</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	64 мегабайта

Рассмотрим прямоугольную таблицу размера $x \times y$, разбитую на единичные клетки. Требуется полностью покрыть эту таблицу одинаковыми фигурами специального вида.

Каждая фигура представляет собой связную конфигурацию из 6 клеток внутри квадрата 3×3 . *Связность* означает, что из любой клетки фигуры можно попасть в любую другую, переходя только по сторонам клеток. Фигура задаётся шаблоном из символов «Н» (занятая клетка) и «о» (пустая клетка).

Все фигуры одинаковы и могут размещаться на поле только в исходной ориентации (запрещено поворачивать и отражать фигуру). Фигуры располагаются так, чтобы их клетки совпадали с клетками таблицы.

Покрытием таблицы называется такое размещение фигур, при котором каждая клетка таблицы покрыта хотя бы одной фигурой. Допускается перекрытие фигур (одна клетка может быть покрыта несколькими фигурами), а также допускается выход частей фигур за границы таблицы.

Требуется определить минимальное количество фигур, необходимое для покрытия всей таблицы.

Формат входных данных

В первой строке заданы два целых числа x и y ($1 \leq x, y \leq 9$) — размеры таблицы.

В следующих трёх строках задан шаблон фигуры размером 3×3 .

Каждая строка содержит ровно три символа двух видов: «Н» — клетка фигуры или «о» — пустая клетка.

Гарантируется, что фигура содержит ровно 6 клеток «Н» и является связной.

Формат выходных данных

Выведите одно число — минимальное количество фигур, необходимое для покрытия всей таблицы.

Система оценки

Каждый тест данной задачи оценивается в $\frac{5}{47}$ балла с округлением до 2-х знаков после запятой по математическим правилам, при этом в случае неверных ответов на произвольный тест задача тестируется до конца по всему набору тестов.

Примеры

table.in	table.out
3 3 oHo HHH HHo	3
3 2 HHH HHH ooo	1
2 3 HHH HHH ooo	2
2 2 HHH HoH ooH	2