

Задача 1. Красивое время

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вова любит подмечать на электронных часах (которые показывают время в формате ЧЧ:ММ, то есть по две цифры для отображения значения часов и минут, например 07:05) «красивые» моменты времени. Красивыми Вова считает такие показания часов, когда выполнено хотя бы одно из следующих условий:

- 1) Значения часов и минут равны, например: 21:21.
- 2) Значения часов и минут выглядят зеркально, то есть значение минут получается перестановкой цифр значений часов в обратном порядке, например: 21:12.
- 3) Значения часов и минут вместе образуют четыре последовательные цифры (то есть каждая следующая цифра на циферблате на единицу больше предыдущей), например: 01:23.

Вова заинтересовался, сколько красивых показаний часов будет на циферблате с момента, когда часы показывают h_1 часов и m_1 минут, до момента, когда часы покажут h_2 часов и m_2 минут (оба показания включительно; если $h_1 = h_2$ и $m_1 = m_2$, то рассматривается одно-единственное показание часов).

Формат входных данных

Во входных данных задачи последовательно записано четыре целых числа: h_1, m_1, h_2, m_2 , каждое число в отдельной строке ($0 \leq h_1, h_2 \leq 23, 0 \leq m_1, m_2 \leq 59$).

Формат выходных данных

В результате работы программы требуется вывести единственное число — количество красивых показаний времени.

Примеры

стандартный ввод	стандартный вывод
20 0 20 30	2
17 31 17 31	0

Задача 2. Сумма делителей

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Найдите натуральное число в диапазоне от 1 до 10^9 , равное сумме каких-то n своих делителей (выбранные делители должны быть различны между собой).

Формат входных данных

Вводится одно целое число n ($1 \leq n \leq 30$).

Формат выходных данных

В первой строке выходных данных выведите искомое число m . Во второй строке выведите n различных натуральных чисел — делителей числа m , сумма которых равна m . Делители можно выводить в любом порядке. Если есть несколько правильных ответов, выведите любой. Если решения нет, выведите -1.

Пример

стандартный ввод	стандартный вывод
3	6 3 2 1

Замечание

Система оценивания: за каждый пройденный тест начисляется некоторое количество баллов, максимальная сумма баллов — 100.

Задача 3. Реклама

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

В некотором государстве решили начать бороться с засилием рекламы на телевизионных каналах и приняли закон, согласно которому в любой произвольно выбранный час суммарное время рекламы не должно превышать T секунд. Напишите программу, которая будет проверять соблюдение этого закона.

Формат входных данных

В первой строке входных данных записано целое число N — количество рекламных вставок за проверяемый период времени ($1 \leq N \leq 10^5$).

Во второй строке записано целое число T ($0 \leq T < 3600$).

В каждой из следующих N строк записаны через пробел два целых числа t_1 и t_2 — время начала и окончания очередной рекламной вставки ($0 \leq t_1 < t_2 \leq 10^9$). Времена задаются в секундах от момента начала проверки. Времена упорядочены по возрастанию (то есть, в каждой следующей строке t_1 больше, чем t_2 в предыдущей строке).

Формат выходных данных

Если найдётся такой час, то есть интервал времени длиной 3600 секунд, в котором было более T секунд рекламы, то выведите одно целое число — время начала такого интервала. Если есть несколько верных ответов, то выведите наименьший неотрицательный из них. Если искомого интервала не существует, то выведите -1.

Примеры

стандартный ввод	стандартный вывод
3 1000 0 100 5000 5300 6000 7000	3101
1 3599 0 3599	-1

Замечание

Система оценивания.

Подзадача 1 (до 30 баллов): $N \leq 100$, $t_2 \leq 10^4$

Подзадача 2 (до 30 баллов): $N \leq 10^5$, $t_2 \leq 10^6$.

Подзадача 3 (до 40 баллов): $N \leq 10^5$, $t_2 \leq 10^9$.

Напоминание для решающих на языке Python: ввести два целых числа, разделённых пробелом, можно так:

```
t1, t2 = map(int, input().split())
```

Задача 4. Зарядка робота

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Робот с кодовым именем МАГД имеет тип n . Для подзарядки ему требуется устройство мощностью p , удовлетворяющее условиям:

$p \geq n + 1$ (иначе роботу не хватит мощности для подзарядки)

$p \leq n + k$ (иначе робот может сгореть)

Из всех подходящих мощностей p нужно выбрать ту, при которой НОК(n, p) минимален, чтобы аккумулятор робота прослужил как можно дольше. По заданным n и k требуется вычислить p .

НОК(a, b) — это наименьшее общее кратное чисел a и b , то есть наименьшее натуральное число, которое делится одновременно и на a , и на b .

Формат входных данных

Вводятся два числа n и k , каждое в отдельной строке ($1 \leq n \leq 10^{12}$, $1 \leq k \leq 10^{12}$)

Формат выходных данных

Требуется вывести число p — искомую мощность устройства.

Примеры

стандартный ввод	стандартный вывод
6 4	9
10 12	20

Замечание

Система оценивания.

Подзадача	Дополнительные условия	Баллы
1	$n \leq 100, k \leq 100$	до 15
2	$n \leq 10^5, k \leq 10^5$	до 15
3	$n \leq 5 \cdot 10^7$	до 30
4	—	до 40

Обратите внимание, что входные данные и ответ могут выходить за пределы диапазона возможных значений 32-битной целочисленной переменной, поэтому необходимо использовать 64-битный целочисленный тип данных (тип `int64` в языке Pascal, тип `long long` в C++, тип `long` в Java и C#). В языке Python ничего дополнительно делать не требуется.

Задача 5. Порядок жизни

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Существует известная математическая игра «Жизнь», правила которой очень просты. Дано клетчатое поле, в котором n строк по m столбцов в каждой. Каждая клетка может иметь одно из двух состояний: она или жива, или мертва. У клетки может быть максимум 8 соседей (если клетка не находится на краях поля). Клетки постоянно конкурируют за существование, и на каждом шаге игры происходят следующие изменения:

1. Живая клетка, имеющая двух или трёх живых соседей, остаётся живой. Иначе она погибает от перенаселения или одиночества.
2. Мёртвая клетка, имеющая трёх живых соседей, оживает.

Игра заканчивается, когда поле перестаёт меняться. Это состояние назовём стабилизацией.

Известно, что на симуляторе игры в «Жизнь» была запущена некоторая первоначальная конфигурация, которая стабилизировалась после некоторого числа шагов. Симулятор зафиксировал положение клеток в k позициях игры, однако, оказался сломан и выдал их в перемешанном порядке.

Вам необходимо определить, в каком порядке в ходе игры появлялись данные конфигурации поля. Гарантируется, что искомая перестановка единственна. Гарантируется, что игра завершится не более чем за 2000 ходов, начав своё существование со стартового положения искомой перестановки.

Формат входных данных

Первая строка содержит одно натуральное число k ($1 \leq k \leq 100$) — количество зафиксированных конфигураций поля.

Вторая строка содержит одно натуральное число n — количество строк поля.

Третья строка содержит одно натуральное число m — количество столбцов. Гарантируется, что $1 \leq n \cdot m \leq 5000$.

Далее идут k описаний зафиксированных конфигураций поля. Каждая конфигурация описывается n строками по m чисел в каждой, разделённых пробелом. Если клетка жива, то она кодируется цифрой 1, иначе — цифрой 0.

Формат выходных данных

Необходимо вывести перестановку натуральных чисел от 1 до k — в каком порядке появлялись данные конфигурации во время игры.

Примеры

стандартный ввод	стандартный вывод
3 3 3 1 0 0 0 1 0 1 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0	1 2 3
2 2 3 1 0 0 1 1 0 1 0 1 1 1 0	2 1

Замечание

Подзадача 1 (5 баллов): $k = 2$, в одной из конфигураций все клетки мёртвы.

Подзадача 2 (до 35 баллов): $k \leq 10$; $n, m \leq 20$.

Подзадача 3 (до 60 баллов): без дополнительных ограничений.

Возможно получение частичных баллов за подзадачи 2 и 3.