

Задача А. Пересадки

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

На Новом проспекте для разгрузки было решено пустить два новых автобусных маршрута на разных участках проспекта. Все остановки на проспекте пронумерованы подряд натуральными числами. Известны конечные остановки каждого из автобусов. Определите количество остановок, на которых можно пересест с одного автобуса на другой.

Формат входного файла

Вводятся четыре натуральных числа — номера конечных остановок сначала первого, потом второго автобуса. Числа не превосходят 10^9 .

Формат выходного файла

Ваша программа должна выводить одно число — искомое количество остановок.

Примеры

<code>transfer.in</code>	<code>transfer.out</code>
3 6 4 2	2
3 1 5 10	0

Задача В. Минимальное покрытие

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

На прямой задано некоторое множество отрезков с целочисленными координатами концов $[L_i, R_i]$. Выберите среди данного множество подмножество отрезков, целиком покрывающее отрезок $[0, M]$, (M — натуральное число), содержащее наименьшее число отрезков.

Формат входного файла

В первой строке указана константа M ($1 \leq M \leq 5000$). В каждой последующей строке записана пара чисел L_i и R_i ($|L_i|, |R_i| \leq 50000$), задающая координаты левого и правого концов отрезков. Список завершается парой нулей. Общее число отрезков не превышает 100 000.

Формат выходного файла

В первой строке выходного файла выведите минимальное число отрезков, необходимое для покрытия отрезка $[0, M]$. Далее выведите список покрывающего подмножества, упорядоченный по возрастанию координат левых концов отрезков. Список отрезков выводится в том же формате, что и во входе. Завершающие два нуля выводить не нужно.

Если покрытие отрезка $[0, M]$ исходным множеством отрезков $[L_i, R_i]$ невозможно, то следует вывести единственную фразу “No solution”.

Примеры

<code>cover.in</code>	<code>cover.out</code>
1 -1 0 -5 -3 2 5 0 0	No solution
1 -1 0 0 1 0 0	1 0 1

Задача С. Объединение отрезков

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

Решая задачу из контрольной по математике, Вася получил ответ в виде объединения N отрезков $[L_i, R_i]$ на числовой прямой. Однако, некоторые из этих отрезков могут пересекаться друг с другом, что не слишком нравится Васе. Ваша задача — представить Васин ответ в виде объединения минимального количества отрезков.

Формат входного файла

В первой строке указано число N ($1 \leq N \leq 50000$). В следующих N строках перечислены пары чисел L_i и R_i ($|L_i|, |R_i| \leq 50000$), каждая пара с новой строки, числа в парах отделены друг от друга одним или несколькими пробелами.

Формат выходного файла

В первой строке выведите число M — количество отрезков в искомом объединении. В следующих M строках выведите сами эти отрезки в том же формате, что и во входном файле. Список отрезков необходимо упорядочить по возрастанию левого конца.

Примеры

<code>merge.in</code>	<code>merge.out</code>
4 0 2 4 5 1 3 5 6	2 0 3 4 6

Задача D. Китайские часы

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

Когда программист Вова был в Китае, он обнаружил, что российские часы «Заря» продаются там в десять раз дешевле, чем в России. Вова решил проверить выгодное дело и закупил огромную партию, с тем чтобы привезти часы на родину и продать их за полцены (то есть, впятеро дороже, чем купил). Но дома обнаружилось, что многие часы идут вразнобой, к тому же, от малейшего толчка часы останавливаются (либо наоборот, начинают идти). Очевидно, это были не настоящие часы фирмы «Заря», а их точные копии. Чтобы быстро продать всю партию часов, Вова хочет поставить их все на одинаковое время (тогда будет важно, правильное это время или нет — можно будет сказать, что это местное время завода-изготовителя) и перед открыванием крышки просто встряхнуть чемодан, чтобы часы одновременно пошли.

Для установки времени на часах есть заводная головка, вращение которой позволяет быстро поворачивать стрелки, при этом часовая стрелка крутится в 60 раз медленнее минутной, а минутная — в 60 раз медленнее секундной. Один оборот заводной головки поворачивает секундную стрелку на полный оборот, и хотя на такой поворот уходит всего одна секунда, чтобы изменить время на 6 часов, потребуется 6 минут. Вращать заводную головку и стрелки можно только по часовой стрелке, чтобы не повредить хрупкий механизм часов. Помогите Вове минимизировать усилия по предпродажной подготовке часов, выбрав время, которое будет установлено на всех часах.

Формат входного файла

В первой строке содержится одно целое число n ($1 \leq n \leq 50000$) — количество часов в партии. Далее следует n строк, $(i + 1)$ -я строка ввода содержит время на i -х часах в формате $h : mm : ss$, где целое число h ($1 \leq h \leq 12$) означает час, а двухразрядные целые числа mm и ss ($00 \leq mm, ss \leq 59$) — минуты и секунды соответственно.

Формат выходного файла

Выведите время, которое нужно установить на всех часах, в формате, указанном выше.

Пример

watches.in	watches.out
3 11:30:00 12:10:01 6:10:18	12:10:01

Задача Е. Кассы

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

На одном из московских вокзалов билеты продают N касс. Каждая касса работает без перерыва определенный промежуток времени по фиксированному расписанию (одному и тому же каждый день). Требуется определить, на протяжении какого времени в течение суток работают все кассы одновременно.

Формат входного файла

Сначала вводится одно целое число N ($0 < N \leq 1000$).

В каждой из следующих N строк через пробел расположены 4 целых числа, первые два из которых обозначают время открытия кассы в часах и минутах (часы — целое число от 0 до 23, минуты — целое число от 0 до 59), оставшиеся два — время закрытия в том же формате. Числа разделены пробелами.

Время открытия означает, что в соответствующую ему минуту касса уже работает, а время закрытия — что в соответствующую минуту касса уже не работает. Например, касса, открытая с 10 ч. 30 мин. до 18 ч. 30 мин., ежедневно работает 480 минут.

Если время открытия совпадает с временем закрытия, то касса работает круглосуточно. Если первое время больше второго, то касса начинает работу до полуночи, а заканчивает — на следующий день.

Формат выходного файла

Требуется вывести одно число — суммарное время за сутки (в минутах), на протяжении которого работают все N касс.

Примеры

tickets.in	tickets.out
3 1 0 23 0 12 0 12 0 22 0 2 0	120
2 9 30 14 0 14 15 21 0	0
2 14 00 18 00 10 00 14 01	1