Общая информация по задачам олимпиады

Доступ к результатам проверки решений задач во время тура

В течение тура по каждой задаче можно отправить не более 40 решений и получить информацию о результатах оценивания решения на тестах жюри.

Требования к программам

Во всех задачах размер файла с исходным кодом решения не должен превышать 256 КБ. В каждой задаче входные данные необходимо считывать из стандартного потока ввода, выходные данные необходимо выводить в стандартный поток вывода.

Процесс тестирования

Перед решением задачи ознакомьтесь с системой оценки решения. Обратите внимание, в некоторых задачах очередная подзадача будет тестироваться, только если пройдены все тесты предыдущих подзадач.

Сложность и порядок задач

Задачи республиканской олимпиады по информатике упорядочены примерно по возрастанию сложности. Полное решение каждой задачи оценивается в 100 баллов.

Ограничения

Задачи	Ограничение по времени	Ограничение по памяти	Получение результатов во время тура
A. Две доминошки	1 секунда	256 МБ	Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи.
B. Потерянные этажи	1 секунда	256 МБ	Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи.
C. Баскетбольный турнир	1 секунда	256 МБ	Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи.
D. Рассадка пассажиров	2 секунды	512 МБ	Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи.
E. Лорды и гербы	1 секунда	256 МБ	Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи.

С результатами проверки решений задач, тестами, решениями жюри, а также письменным разбором задач можно ознакомиться после окончания тура на сайте https://kpfu.ru/math/dlya-shkolnikov/put-k-olimpu-informatika

Задача A. Две доминошки

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

Доминошка представляет собой прямоугольную плитку размером 1×2, разделенную на две половинки. На каждой из них нарисовано от 0 до 6 точек. По правилам игры две доминошки можно поставить рядом, если у них есть половинки с одинаковым числом точек. Например, доминошки $\boxed{1}\boxed{2}$ и $\boxed{1}\boxed{4}$, а также $\boxed{0}\boxed{1}$ и $\boxed{5}\boxed{1}$ можно поставить рядом, а доминошки $\boxed{1}\boxed{2}$ и $\boxed{3}\boxed{4}$ — нельзя. (Числа означают количества точек на половинках доминошек.)

Вам необходимо для заданной пары доминошек определить, можно ли их поставить рядом друг с другом.

Формат входных данных

В первой строке записаны через пробел два целых числа a и b — количество точек на половинках первой доминошки ($0 \leqslant a, b \leqslant 6$). Во второй строке записаны через пробел два целых числа c и d — количество точек на половинках второй доминошки ($0 \leqslant c, d \leqslant 6$). Гарантируется, что доминошки отличаются количеством точек хотя бы на одной из половинок.

Формат выходных данных

Выведите число -1, если доминошки нельзя поставить рядом друг с другом. В противном случае запишите через пробел исходные числа a, b, c, d в порядке расположения доминошек по правилам игры. Если решений несколько, выведите любое из них.

Система оценки

В этой задаче 25 тестов. Каждый тест оценивается в 4 балла.

Примеры

стандартный ввод	стандартный вывод
0 1 1 4	0 1 1 4
0 1 5 1	0 1 1 5
1 2 3 4	-1
0 1 1 4	0 1 1 4
0 1 5 1	0 1 1 5
1 2 3 4	-1

Задача B. Потерянные этажи

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

Городская администрация провела полный учёт этажности всех жилых зданий. Для каждого целого числа k от 0 до $m - 1$ было подсчитано количество домов a_k, имеющих **более k этажей**. Таким образом, получена последовательность $a_0, a_1, \ldots, a_{m-1}$.

Эти сводные данные были переданы в аналитический отдел для дальнейшей обработки. Однако исходная информация о каждом доме по отдельности была утрачена. Теперь необходимо восстановить, сколько этажей было в каждом конкретном здании, используя только имеющуюся статистику.

Требуется написать программу, которая по заданным значениям $a_0, a_1, \ldots, a_{m-1}$ восстановит количество этажей в каждом доме города и выведет их в порядке невозрастания.

Формат входных данных

В первой строке записано одно целое число m — количество элементов в последовательности ($1 \leqslant m \leqslant 10^6$). Во второй строке записаны m целых чисел $a_0, a_1, \ldots, a_{m-1}$, разделённых пробелами ($1 \leqslant a_i \leqslant 10^6$). Гарантируется, что последовательность (a_i) является невозрастающей, то есть $a_0 \geqslant a_1 \geqslant \ldots \geqslant a_{m-1} > 0$.

Формат выходных данных

Выходные данные должны содержать две строки.

В первой строке выведите одно целое число n — количество домов в городе.

Во второй строке запишите n целых чисел в порядке невозрастания — количество этажей в каждом доме.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
1	40	$1 \leqslant m \leqslant 40\,000$		баллы
2	40	$1 \leqslant m \leqslant 10^5$	1	баллы
3	20	$1 \leqslant m \leqslant 10^6$	1, 2	баллы

Пример

стандартный ввод	стандартный вывод
5 4 4 3 1 1	4 5 3 3 2

Задача C. Баскетбольный турнир

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

В турнире по баскетболу без ничьих участвовало n команд. Каждая игра завершалась победой одной из команд. Арсений выписал количество побед каждой команды, но не знает, по какой системе проводился турнир. У него есть два предположения:

- Круговая система: каждая команда играет с каждой ровно один раз.
- Олимпийская система: турнир на выбывание, где команда выбывает после первого проигрыша.

Помогите Арсению определить, какая система могла быть использована, исходя из известных количеств побед команд.

Формат входных данных

В первой строке записано целое число n ($3 \leqslant n \leqslant 10^5$) — количество команд.

В каждой из следующих n строк записаны название команды и число побед.

Гарантируется, что сумма длин названий команд не больше $3 \cdot 10^5$.

Формат выходных данных

Выведите `Round-robin` если команды играли круговой турнир, иначе `Olympic`.

Система оценки

В задаче 20 тестов, каждый тест оценивается в 5 баллов.

Примеры

стандартный ввод	стандартный вывод
3 aba 1 abc 2 rty 0	Round-robin
4 a 0 b 0 c 1 d 2	Olympic

Задача D. Рассадка пассажиров

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Началась посадка пассажиров на борт самолёта. В самолёте n рядов, в каждом ряду по m кресел. Кресло с номером 1 находится у окна, а кресло с номером m — у прохода.

Задан порядок, в котором пассажиры заходят в самолёт, а также место каждого пассажира. Когда очередной пассажир должен занять место с номером y в своём ряду, он проходит от прохода к своему месту. Если на каком-либо кресле с номером t, где $y < t \leqslant m$ (то есть между проходом и его местом), уже сидит другой пассажир, то этот пассажир обязан встать, пропустить проходящего и затем снова сесть. Каждое такое действие считается одним вставанием.

Всего на рейс зарегистрировано k пассажиров. Пассажиры заходят в самолёт по очереди, в порядке от 1 до k. Требуется вычислить общее количество вставаний, произошедших при рассадке всех пассажиров по местам.

Формат входных данных

В первой строке заданы три целых числа n, m и k ($1 \leqslant n \leqslant 10^9$, $1 \leqslant m \leqslant 10^9$, $1 \leqslant k \leqslant 10^5$) — количество рядов, количество мест в каждом ряду и количество пассажиров соответственно. Гарантируется, что $k \leqslant n \cdot m$.

Далее идут k строк. В i-й строке заданы два целых числа x_i и y_i ($1 \leqslant x_i \leqslant n$, $1 \leqslant y_i \leqslant m$) — номер ряда и номер места в ряду, на которое должен сесть i-й пассажир.

Гарантируется, что все заданные места попарно различны.

Формат выходных данных

Выведите одно целое число — общее количество вставаний пассажиров при рассадке.

Система оценки

Подзадача	**Баллы**	**Ограничения**	**Необходимые подзадачи**
0	0	Примеры из условия	—
1	15	$1 \leqslant n \leqslant 100$ $m = 3$	—
2	15	$1 \leqslant n \leqslant 100$ $1 \leqslant m \leqslant 100$	1
3	20	$1 \leqslant n \leqslant 10^9$ $1 \leqslant m \leqslant 100$	1, 2
4	25	$1 \leqslant n \leqslant 10^9$ $1 \leqslant m \leqslant 10^5$	1, 2, 3
5	25	Основные ограничения	1, 2, 3, 4

Примеры

стандартный ввод	стандартный вывод
1 2 2 1 2 1 1	1
1 2 2 1 1 1 2	0
2 2 2 1 2 2 1	0
2 3 5 1 3 2 2 2 3 1 1 2 1	3
3 3 9 1 3 1 2 1 1 2 2 2 1 2 3 3 1 3 3 3 2	5

Задача E. Лорды и гербы

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

В английском королевстве живут n лордов. Каждый лорд имеет несколько родовых гербов (например, фамильный герб, герб супруги или гербы присоединённых владений). Лорды пронумерованы целыми числами от 1 до n, а гербы — целыми числами от 1 до n.

Два лорда могут быть представлены друг другу напрямую, если у них есть общий родовой герб. В противном случае представление может состояться через цепочку других лордов-посредников: лорд A представлен лорду p_1 по общему гербу, лорд p_1 представлен лорду p_2 по другому общему гербу, и так далее, пока лорд p_k не будет представлен лорду B.

Требуется определить наименьшее количество посредников, необходимое для того, чтобы лорд A мог быть представлен лорду B.

Формат входных данных

Первая строка содержит число n ($2 \leqslant n \leqslant 1\,500$) — количество лордов.

В каждой из следующих n строк записаны: число g_i ($1 \leqslant g_i \leqslant n-1$) — количество родовых гербов i-го лорда, а затем через пробел g_i различных целых чисел — номера этих гербов. Гарантируется, что общая сумма гербов не превосходит 300 000.

В последней $(n+2)$-ой строке записаны через пробел номера лордов A и B — различные натуральные числа, не превосходящие n.

Формат выходных данных

Выведите -1, если представление между лордами A и B невозможно.

Выведите 0, если у A и B есть общий герб (представление без посредников).

Иначе в первой строке выведите натуральное число k — минимальное количество посредников. Во второй строке выведите через пробел номера посредников $p_1, p_2, \ldots, p_k$ в порядке представления от A к B. Если решений несколько, выведите любое.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
1	20	$2 \leqslant n \leqslant 30$, $\sum g_i \leqslant 3\,000$		баллы
2	25	$2 \leqslant n \leqslant 200$, $\sum g_i \leqslant 20\,000$	1	баллы
3	25	$2 \leqslant n \leqslant 1\,000$, $\sum g_i \leqslant 150\,000$	1, 2	баллы
4	30	$2 \leqslant n \leqslant 1\,500$, $\sum g_i \leqslant 300\,000$	1, 2, 3	баллы

Примеры

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