

Общая информация по задачам второго тура

Доступ к результатам проверки решений задач во время тура

В течение тура можно не более 10 раз по каждой задаче запросить информацию о результатах оценивания решения на тестах жюри. Запрос по каждой задаче можно делать не чаще одного раза в 5 минут.

Ограничение на размер исходного кода программы-решения

Во всех задачах размер файла с исходным кодом решения не должен превышать 256 КБ.

Ограничения

Задача	Ограничение по времени	Ограничение по памяти	Получение результатов во время тура
5. Киноакадемия	1 секунда	256 МБ	Для каждой подзадачи сообщаются баллы за эту подзадачу и результат проверки программы на каждом тесте
6. Съезд кинозвезд	Задача с открытыми тестами		Для каждого теста сообщаются баллы за этот тест и комментарий проверяющей программы
7. Здоровое питание	2 секунды	256 МБ	Сообщается результат проверки программы на каждом тесте
8. Магистраль «Урал» 2.0	2 секунды	256 МБ	Для каждой подзадачи сообщаются баллы за эту подзадачу и результат проверки программы на каждом тесте

Задача 5. Киноакадемия

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

В финал конкурса Киноакадемии вышли n лучших кинофильмов 2014 года. В конкурсе награждаются фильмы в двух номинациях: лучшая режиссура и лучший сценарий. По правилам конкурса в каждой номинации должен быть награжден ровно один фильм, причём в разных номинациях — разные фильмы.

В ходе многочисленных опросов зрителей и кинокритиков удалось собрать данные, показывающие, какой уровень ликования вызовет победа каждого фильма в каждой из номинаций. Дотошные журналисты на этом не остановились и дополнительно выяснили, каким будет уровень ликования, если тот или иной фильм не выиграет ни в одной из номинаций.

Требуется написать программу, которая по результатам опросов определяет наибольший суммарный уровень ликования, которого можно добиться выбором фильмов для награждения в указанных номинациях.

Формат входных данных

В первой строке входного файла задано целое число n — количество кинофильмов, участвующих в финале конкурса Киноакадемии. В следующих n строках содержатся по три целых числа a_i, b_i, c_i — уровень ликования, если i -й фильм не выиграет ни в одной из номинаций, уровень ликования, если этот фильм выиграет в номинации на лучшую режиссуру, и уровень ликования, если этот фильм выиграет в номинации на лучший сценарий.

Формат выходных данных

Первая строка выходного файла должна содержать одно число — наибольший возможный суммарный уровень ликования. Вторая строка должна содержать два целых числа — номера фильмов-победителей в номинациях лучшая режиссура и лучший сценарий соответственно. Фильмы нумеруются натуральными числами от 1 до n . Если оптимальных способов выбора награждаемых фильмов несколько, можно вывести любой из них.

Система оценки

Данная задача содержит три подзадачи. Для оценки каждой подзадачи используется своя группа тестов. Баллы за подзадачу начисляются только в том случае, если все тесты из этой группы пройдены.

Подзадача 1

$$2 \leq n \leq 100$$
$$1 \leq a_i, b_i, c_i \leq 10^5$$

Подзадача оценивается в 20 баллов.

Подзадача 2

$$2 \leq n \leq 2000$$
$$1 \leq a_i, b_i, c_i \leq 10^5$$

Подзадача оценивается в 25 баллов.

Подзадача 3

$$2 \leq n \leq 10^5$$
$$1 \leq a_i, b_i, c_i \leq 10^9$$

Подзадача оценивается в 55 баллов.

Пример

cinema.in	cinema.out
3	17
3 6 9	2 3
1 5 7	
1 3 9	

Пояснение к примеру

В приведенном примере наибольший суммарный уровень ликования равен $3 + 5 + 9 = 17$.

Задача 6. Съезд кинозвёзд

На съезд лауреатов конкурсов Киноакадемии приглашены n кинозвёзд, которые очень трепетно относятся к различным слухам о себе. Необходимо, чтобы среди $\frac{n(n-1)}{2}$ возможных пар кинозвёзд оказалось ровно a пар, в которых обе кинозвезды ни в какой момент времени не будут присутствовать в зале съезда вместе, и ровно b пар, в которых одна из кинозвёзд будет присутствовать в зале только вместе с другой: войдет в зал позже нее, а выйдет раньше.

Чтобы обеспечить эти условия, на входе в зал поставили швейцара. В каждый момент времени он либо впускает одного человека в зал, либо выпускает одного человека из зала. Кинозвёздам, покинувшим зал, запрещено вновь туда возвращаться.

Требуется для каждого из q заданных съездов по n , a и b определить подходящую последовательность входа и выхода кинозвёзд в зал.

Формат входных данных

Первая строка входного файла содержит целое число q — количество съездов. Каждая из последующих q строк содержит описание съезда: три целых числа n , a и b .

Формат выходных данных

Выходной файл должен содержать q строк — по одной на каждый съезд. Каждая строка должна содержать число n , после которого следуют $2n$ целых чисел, описывающих порядок входа и выхода кинозвёзд в зал. Каждое число в диапазоне от 1 до n должно встречаться дважды: в первый раз число i обозначает вход i -й кинозвёзды в зал, во второй раз — её выход.

Гарантируется, что для каждого из заданных съездов существует хотя бы одно решение. Если существует несколько решений, можно вывести любое из них.

Если решение для конкретного съезда вами не найдено, в соответствующей строке необходимо вывести единственное число 0.

Система оценки

Требуется решить задачу для 7 тестов, которые находятся на вашем компьютере в каталоге «с:\work\6-tests» в файлах с именами 01, 02, 03, 04, 05, 06 и 07. На проверку требуется сдать только файлы с ответами. Сдавать программу не требуется.

За каждый тест, ответ на который отправляется в проверяющую систему, начисляется k баллов, где k — число съездов, для которых было найдено правильное решение.

Если присланный файл не соответствует требованиям к формату выходных данных, то он не будет принят на окончательную проверку с сообщением «PE 1».

Доступ к результатам проверки во время тура

В течение тура можно не более 10 раз по каждому тесту запросить информацию о результатах оценивания ответа. Запрос по каждому тесту можно делать не чаще одного раза в 5 минут. В качестве результата проверки сообщается количество полученных баллов и комментарий проверяющей программы, который содержит информацию о правильности решений для съездов в порядке их перечисления во входном файле в следующем формате:

- «+» — решение правильное, за него начислен 1 балл;
- «-» — решение неправильное;
- «0» — присланный файл содержит 0 для этого съезда.

Пример

тест	ответ	комментарий проверяющей программы
4	3 1 2 3 3 2 1	2 из 4 тестов пройдено + 0 + -
3 0 3	0	
3 0 0	3 1 1 2 2 3 3	
3 3 0	3 1 2 3 3 2 1	
3 3 0		

Пояснение к примеру

В приведённом в примере ответе на тест, решение для второго съезда не найдено, а решение для четвертого теста неправильное. Если бы это был один из тестов жюри, такой ответ был бы оценен в 2 балла из 4.

Задача 7. Здоровое питание

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

План студенческого городка некоторого университета представляет собой квадрат $n \times n$, в каждой клетке которого расположено здание. Здания соединены переходами, если они расположены в клетках, имеющих общую сторону. В левом верхнем углу квадрата расположено студенческое общежитие. В правом нижнем углу расположен учебный корпус.

В каждом из зданий, включая общежитие и учебный корпус, расположен автомат, торгующий ровно одним продуктом, например, только кофе или только пирожками с мясом. Студенты каждый день ходят из общежития в учебный корпус по переходам, выбирая один из кратчайших путей.

Руководство университета заинтересовалось разнообразием питания студентов, покупающих продукты в автоматах по ходу движения. Для каждого автомата $A_{i,j}$ планируется найти кратчайший путь из общежития в учебный корпус, проходящий через этот автомат и содержащий как можно больше автоматов, торгующих тем же самым продуктом, что и автомат $A_{i,j}$. Количество таких автоматов на этом пути называется *избыточностью* автомата $A_{i,j}$. При этом автомат $A_{1,1}$ находится в общежитии, а автомат $A_{n,n}$ — в учебном корпусе.

Требуется написать программу, которая по информации о продуктах, продаваемых автоматами, для каждого из чисел в диапазоне от 1 до $2n - 1$ определяет число автоматов с таким значением избыточности.

Формат входных данных

Первая строка входного файла содержит целое число n ($2 \leq n \leq 1500$). Следующие n строк содержат по n чисел в каждой. В i -й из этих строк j -е число соответствует номеру продукта, продающегося в автомате $A_{i,j}$. Номера продуктов находятся в диапазоне от 1 до n^2 .

Формат выходных данных

Выходной файл должен содержать $(2n - 1)$ целых чисел — количество автоматов с избыточностями $1, 2, \dots, 2n - 1$ соответственно.

Система оценки

Для проверки решений этой задачи используются 50 тестов. Тесты оцениваются независимо. Каждый тест оценивается в 2 балла. Значения n в тестах жюри приведены в следующей таблице.

Тест	Значение n	Тест	Значение n	Тест	Значение n	Тест	Значение n	Тест	Значение n
1.	$n = 2$	11.	$n = 50$	21.	$n = 200$	31.	$n = 550$	41.	$n = 1050$
2.	$n = 4$	12.	$n = 60$	22.	$n = 225$	32.	$n = 600$	42.	$n = 1100$
3.	$n = 6$	13.	$n = 70$	23.	$n = 250$	33.	$n = 650$	43.	$n = 1150$
4.	$n = 8$	14.	$n = 80$	24.	$n = 275$	34.	$n = 700$	44.	$n = 1200$
5.	$n = 10$	15.	$n = 90$	25.	$n = 300$	35.	$n = 750$	45.	$n = 1250$
6.	$n = 15$	16.	$n = 100$	26.	$n = 325$	36.	$n = 800$	46.	$n = 1300$
7.	$n = 20$	17.	$n = 120$	27.	$n = 350$	37.	$n = 850$	47.	$n = 1350$
8.	$n = 25$	18.	$n = 140$	28.	$n = 400$	38.	$n = 900$	48.	$n = 1400$
9.	$n = 30$	19.	$n = 160$	29.	$n = 450$	39.	$n = 950$	49.	$n = 1450$
10.	$n = 40$	20.	$n = 180$	30.	$n = 500$	40.	$n = 1000$	50.	$n = 1500$

Примеры

<code>fastfood.in</code>	<code>fastfood.out</code>
3 1 1 1 2 2 2 3 3 3	0 0 9 0 0
5 1 4 1 3 5 2 1 4 1 2 5 1 1 4 5 3 5 1 1 2 4 3 5 1 1	2 4 9 0 0 1 1 8 0

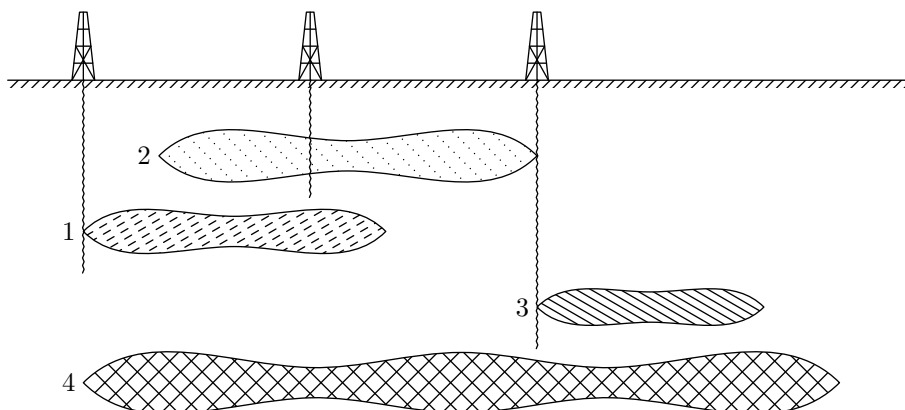
Задача 8. Магистраль «Урал»

Имя входного файла:	drill.in
Имя выходного файла:	drill.out
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 МБ

Планируется строительство новой магистрали «Урал». Долговечность автомагистрали зависит от пластов пород, залегающих под ней. Пластом называется геологическое тело, состоящее из одной горной породы.

Под будущей магистралью залегают n горизонтальных пластов. Геологическое исследование позволило определить точки магистрали, под которыми начинается и заканчивается каждый из них. При этом порядок залегания пластов по глубине определить не удалось.

В заданных местах вдоль планируемой магистрали пробурены вертикальные скважины. Каждая из них пересекает несколько верхних пластов, находящихся под точкой бурения. Для каждой скважины известно, в каком порядке располагаются пробуренные пласты сверху вниз, начиная от поверхности. Если скважина не пересекает какой-то из пластов, находящихся под точкой бурения, значит он проходит ниже дна скважины.



Требуется написать программу, которая определяет возможный порядок залегания пластов по глубине, не противоречащий полученным данным.

Формат входных данных

Первая строка входного файла содержит целое число n — количество пластов. Пласты пронумерованы целыми числами от 1 до n в произвольном порядке.

В i -й из следующих n строк содержатся целые числа l_i и r_i ($0 \leq l_i < r_i \leq 10^9$) — расстояния от начала магистрали до точек, под которыми начинается и заканчивается i -й пласт.

В следующей строке записано целое число m — количество скважин, в которых проводилось бурение. Следующие m строк описывают результаты бурения: в каждой строке сначала указаны два целых числа x ($0 \leq x \leq 10^9$) и k ($0 \leq k \leq n$) — расстояние от начала магистрали до скважины и количество обнаруженных в данной скважине пластов, затем — целые числа s_1, s_2, \dots, s_k — номера пробуренных пластов, перечисленные в порядке залегания сверху вниз. Скважины перечислены в порядке возрастания расстояния x .

Гарантируется, что решение существует.

Формат выходных данных

Первая строка выходного файла должна содержать n целых чисел p_1, p_2, \dots, p_n , описывающих возможный порядок залегания пластов сверху вниз. Среди чисел p_1, p_2, \dots, p_n каждый номер пласта должен встретиться ровно один раз. При этом пласт с номером p_j не должен нигде проходить выше пластов с номерами p_1, \dots, p_{j-1} или ниже пластов с номерами p_{j+1}, \dots, p_n .

Если возможных расположений пластов несколько, выведите любое из них.

Система оценки

Данная задача содержит пять подзадач. Для оценки каждой подзадачи используется своя группа тестов. Баллы за подзадачу начисляются только в том случае, если все тесты из этой группы пройдены.

Подзадача 1

$1 \leq n, m \leq 1000$

Каждая скважина пересекает все пласты, залегающие под ней.

Подзадача оценивается в 20 баллов.

Подзадача 2

$$1 \leq n, m \leq 1000.$$

Подзадача оценивается в 20 баллов.

Подзадача 3

$$1 \leq n, m \leq 30\,000.$$

Суммарное количество пластов, найденных при бурении скважин, не более 10^6 .

Подзадача оценивается в 20 баллов.

Подзадача 4

$$1 \leq n, m \leq 10^5.$$

Суммарное количество пластов, найденных при бурении скважин, не более 10^5 .

Подзадача оценивается в 20 баллов.

Подзадача 5

$$1 \leq n, m \leq 10^5.$$

Суммарное количество пластов, найденных при бурении скважин, не более 10^6 .

Подзадача оценивается в 20 баллов.

Пример

drill.in	drill.out
4	2 1 3 4
1 5	
2 7	
7 10	
1 11	
3	
1 1 1	
4 1 2	
7 2 2 3	

Пояснение к примеру

Рисунок в условии соответствует примеру.

Для приведенного примера правильным также является ответ 2 3 1 4.

Обратите внимание, что тест из примера не соответствует подзадаче 1. Для того, чтобы решение было принято на проверку, оно должно проходить тест из примера, даже если решена только эта подзадача.