



Общая информация по задачам олимпиады

Ограничение по памяти

Во всех задачах ограничение составляет 512 МБ.

Ограничение на размер исходного кода программы

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

Ограничение на посылку решений

По каждой задаче на проверку принимается не более 50 решений.

По каждой задаче участник не может отправить решение более одного раза в течение 30 секунд. Это ограничение не распространяется на последние 15 минут соревнований.

Система оценки

Каждая задача олимпиады поделена на несколько подзадач. Чтобы набрать баллы по подзадаче, программа должна пройти все тесты этой подзадачи.

За каждую задачу выставляется суммарный балл по всем ее подзадачам. В каждой подзадаче оценивается лучшее решение, то есть за подзадачу выставляется максимальный набранный по ней балл среди всех решений.

Получение информации о результатах проверки

Чтобы получить информацию о проверке вашего решения, используйте ссылку «Информация о проверке» во вкладке «Решения» в PCMS2 Web Client. По каждой задаче вам будет доступна информация по количеству набранных баллов в каждой подзадаче или результат проверки на первом непройденном тесте.

Таблица результатов

Во время соревнования доступна текущая таблица результатов. Для доступа к ней используйте ссылку «Результаты» в PCMS2 Web Client. Таблица результатов в PCMS2 Web Client не является окончательной.



Задача A. The Battle of Giants

Ограничение по времени: 1 секунда

Известный организатор турниров по спортивному программированию решил провести соревнование чемпионов «Битва гигантов». В битве участвуют две команды. Командам устраивается несколько матчей. Результатом каждого матча может стать либо победа одной из команд, либо ничья. При победе команда получает 3 очка, а ее соперник не получает ни одного очка. При ничье обе команды получают по 1 очку. По результатам всех матчей подсчитывается общий счет $a:b$, где a и b — число очков, набранное первой и второй командой, соответственно. Например, если в первом матче победила первая команда, второй матч завершился ничьей, а в третьем матче также победила первая команда, общий счет будет 7:1.

Вам задан общий счет битвы. Определите мог ли такой общий счет получиться, и при каком минимальном числе матчей такой счет мог получиться. Выведите при этом, сколько матчей выиграла первая команда, сколько матчей завершились в ничью, и сколько матчей выиграла вторая команда.

Формат входных данных

В первой строке задано целое число a — суммарное число очков, набранное первой командой.

Во второй строке задано целое число b — суммарное число очков, набранное второй командой.

Числа a и b неотрицательны и не превосходят 10^9 .

Формат выходных данных

Выведите одно число -1 , если заданный общий счет не мог быть получен в результате битвы.

Если же заданный общий счет может быть получен, выведите три целых числа: число побед первой команды, число ничьих и число побед второй команды, соответственно. Требуется найти любой ответ, при котором число матчей минимально.

Система оценки

Подзадача	Баллы	Ограничения
1	21	$0 \leq a \leq 50; b = 0$
2	23	$0 \leq a, b \leq 50; a = b$
3	25	$0 \leq a, b \leq 50$
4	31	$0 \leq a, b \leq 10^9$

Примеры

стандартный ввод	стандартный вывод
7 1	2 1 0
2 1	-1



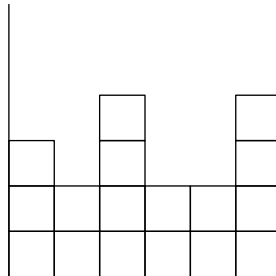
Задача В. Tetris Remastered

Ограничение по времени: 1 секунда

Мила любит играть в Тетрис. Сегодня ей попала новая игра, которая очень похожа на Тетрис. В этой игре есть поле в форме стакана ширины n , разделенное на клетки размером 1×1 . В отличие от обычного Тетриса, в этой игре используются горизонтальные фигурки $1 \times x$, состоящие из x клеток: высоты 1 и ширины x . Перед падением очередной фигурки, игрок может выбрать ее размер x любым целым числом от 1 до n , включительно. Фигурки нельзя поворачивать, но можно двигать влево и вправо. Фигурка падает до тех пор, пока не наткнется на другую фигурку, либо на дно стакана.

Мила не любит оставлять пустые клетки под фигурами. Ее цель — заполнить нижние ряды поля, чтобы занятая фигурками часть образовала прямоугольник ширины n .

Вам задано состояние поля в формате: a_1, a_2, \dots, a_n , где a_i — число клеток, занятых в i -м столбце стакана. В заданном поле никакая пустая клетка не находится под занятой. Например, для последовательности a , равной 3, 2, 4, 2, 2, 4, поле будет выглядеть так:



Найдите, какое минимальное число фигурок ей понадобится, чтобы Мила смогла заполнить нижние ряды поля, образовав прямоугольник ширины n .

Формат входных данных

В первой строке задано целое число n — ширина игрового поля ($1 \leq n \leq 2 \cdot 10^5$).

Во второй строке заданы n целых чисел a_1, a_2, \dots, a_n — число занятых клеток в каждом из столбцов поля ($0 \leq a_i \leq 10^9$).

Хотя бы одно из a_i больше 0.

Формат выходных данных

Выведите единственное целое число: минимальное число фигурок, необходимое Миле для составления прямоугольника ширины n .

Система оценки

Подзадача	Баллы	Ограничения	
		n	a_i
1	8	$n \leq 10$	$a_i \leq 5$
2	13	$n \leq 100$	$a_i \leq 500$
3	16	$n \leq 1000$	$a_i \leq 5000$
4	17	$n \leq 1000$	$a_i \leq 10^9$
5	25	$n \leq 10^5$	$a_i \leq 10^9$ при $n > 1000$ последовательность a сгенерирована случайно
6	21	$n \leq 2 \cdot 10^5$	$a_i \leq 10^9$

Пример

стандартный ввод	стандартный вывод
6 3 2 4 2 2 4	4



Пояснение к примеру

В примере Мила может использовать следующие четыре фигурки:

3		4			
	2		1		



Задача С. Optimal Truck

Ограничение по времени: 2 секунды

Петя собирается купить грузовичок и заняться перевозками. Он изучил рынок и выяснил, что в его городе есть n потенциальных клиентов. Для i -го клиента есть m_i вариантов контракта. Каждый вариант задается двумя числами: w_{ij} — минимальная грузоподъемность грузовика, которая необходима, чтобы выполнить контракт, и c_{ij} — прибыль, которую получит Петя, если заключит этот контракт. С каждым клиентом можно заключить не более одного контракта.

Сейчас Петя думает, какой грузовик лучше купить, чтобы получить нужную ему прибыль. У него есть q вариантов. В i -м варианте Петя хочет, чтобы его прибыль была не меньше x_i . Помогите ему для каждого из вариантов найти минимальную грузоподъемность грузовика, с которой можно получить такую прибыль.

Формат входных данных

Первая строка содержит число n ($1 \leq n \leq 10^5$).

Далее следуют n блоков, описывающих варианты контракта для каждого из потенциальных клиентов. Каждый такой блок начинается с числа m_i , далее следуют m_i пар чисел w_{ij}, c_{ij} ($1 \leq m_i, \sum m_i \leq 5 \cdot 10^5, 1 \leq w_{ij}, c_{ij} \leq 10^9$).

Далее следует число q ($1 \leq q \leq 10^5$). Далее следуют q чисел x_i ($1 \leq x_i \leq 10^9$).

Формат выходных данных

Выведите q чисел — минимальную грузоподъемность грузовика, с которой можно получить требуемую прибыль, для каждого из вариантов. Если требуемую прибыль получить невозможно, выведите -1 для соответствующего варианта.

Система оценки

Подзадача	Баллы	Ограничения			
		n	$\sum m_i$	q	Дополнительно
1	7	$n \leq 5$	$\sum m_i \leq 10$	$q \leq 10$	$w_i, c_i, x_i \leq 100$
2	15	$n \leq 100$	$\sum m_i \leq 500$	$q \leq 10$	$w_i, c_i, x_i \leq 100$
3	21	$n \leq 10^5$	$\sum m_i \leq 5 \cdot 10^5$	$q \leq 10$	—
4	24	$n \leq 10^5$	$\sum m_i \leq 5 \cdot 10^5$	$q \leq 10^5$	$w_i \leq 100$
5	33	$n \leq 10^5$	$\sum m_i \leq 5 \cdot 10^5$	$q \leq 10^5$	—

Пример

стандартный ввод	стандартный вывод
3	1 40 20 -1 10
2	
10 20	
20 30	
1	
40 50	
3	
2 5	
1 10	
4 7	
5	
10 55 32 100 17	



Задача D. Bookshelf Sorting

Ограничение по времени: 2 секунды

Ирма работает в библиотеке. Каждый день она наблюдает, как посетители берут пару книг с полки, читают их, затем ставят на те же места, откуда они их взяли, но не в том же порядке, в котором они стояли изначально. Рассмотрим одну конкретную полку, на которой стоят n книг в некотором порядке, позиции книг пронумерованы слева направо числами от 1 до n . Будем считать, что i -й посетитель берет книги на позициях x_i и y_i , читает их, и ставит их обратно, но в неправильном порядке, то есть книга, ранее стоявшая на позиции x_i теперь будет стоять на позиции y_i , и наоборот.

Вечером, после закрытия, Ирма хочет вернуть книги на свои места. Для каждой книги она знает число p_i — позицию, на которой книга должна стоять в конечном итоге. Для того, чтобы поставить книги в желаемом порядке, Ирма может вытащить любую книгу, и поставить ее либо в начало, либо в конец полки (на первую или последнюю позицию соответственно).

Какое минимальное число таких действий нужно совершить Ирме, чтобы упорядочить книги? Ответьте на этот вопрос для некоторого начального расположения книг, заданного числами p_i , а также после каждого посетителя, который поменял некоторые две книги местами.

Формат входных данных

Первая строка содержит два целых числа n и q ($2 \leq n \leq 2 \cdot 10^5$; $0 \leq q \leq 2 \cdot 10^5$) — количество книг на полке и количество посетителей. Следующая строка содержит n различных целых чисел p_i ($1 \leq p_i \leq n$), означающие, что книга на i -й позиции должна оказаться на позиции p_i .

Следующие q строк описывают действия посетителей. Каждая строка содержит пару целых чисел x_i, y_i ($1 \leq x_i < y_i \leq n$), и означает то, что i -й посетитель поменял местами книги на позициях x_i и y_i .

Формат выходных данных

Выведите $q + 1$ целое число — минимальное количество действий, необходимое для того, чтобы упорядочить все книги на полке для изначальной расстановки, после первого посетителя, ..., после всех q посетителей.

Система оценки

Подзадача	Баллы	Ограничения
1	10	$n, q \leq 8$
2	15	$n, q \leq 200$
3	15	$n \leq 2000$; $q = 0$
4	15	$n, q \leq 2000$
5	20	$n \leq 2 \cdot 10^5$; $q = 0$
6	25	$n, q \leq 2 \cdot 10^5$

Пример

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

Замечание

Изначально книги стоят в порядке (5, 1, 2, 4, 3). Для того, чтобы поставить книги на свои места, нужно сначала вытащить книгу с номером 4 и поставить ее в конец. Затем сделать то же самое с книгой 5. После первого изменения порядок выглядит как (5, 1, 2, 3, 4); здесь достаточно книгу 5 поставить на последнее место. После второго изменения порядок книг становится (3, 1, 2, 5, 4). В этом случае минимальное количество действия равно 3, существует несколько способов добиться верного порядка за 3 действия.



Задача E. Nice Shape

Ограничение по времени: 4 секунды

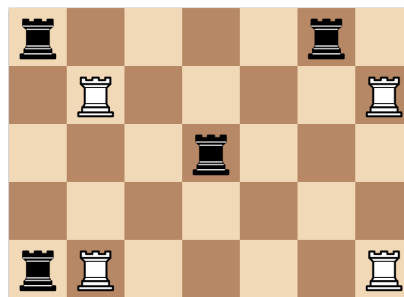
Вам даны n ладей на бесконечной шахматной доске.

i -я из них находится в клетке (r_i, c_i) .

За один ход вы можете переместить любую ладью на любую клетку в той же строке/столбце. Иначе говоря, за один ход вы можете выбрать любое i а затем либо заменить r_i на любое целое число, либо заменить c_i на любое целое число. Вы не можете двигать ладью в клетку, которая содержит другую ладью.

Четыре разные ладьи a, b, c, d образуют *симпатичную фигуру* если вы можете найти прямоугольник, что a, b, c, d это его углы. Иначе говоря, если множество клеток $\{(r_a, c_a), (r_b, c_b), (r_c, c_c), (r_d, c_d)\}$ совпадает с множеством $\{(x_1, y_1), (x_1, y_2), (x_2, y_1), (x_2, y_2)\}$ для какого-то набора целых чисел x_1, x_2, y_1, y_2 , где $x_1 \neq x_2$ и $y_1 \neq y_2$

Например, белые ладьи на следующей картинке образуют симпатичную фигуру.



Ваша задача — найти минимальное количество действий, которое нужно совершить, чтобы получить симпатичную фигурку.

Иначе говоря, вам нужно найти минимальное число операций, после которого может найтись прямоугольник, который содержит ладьи в четырех углах.

Формат входных данных

В первой строке записано одно целое число t ($1 \leq t \leq 25\,000$): количество наборов входных данных.

Далее следуют описания t наборов входных данных.

В первой строке записано одно целое число n ($4 \leq n \leq 100\,000$).

В i -й из следующих n строк записаны два целых числа r_i, c_i ($1 \leq r_i, c_i \leq 10^9$)

Для каждой пары i, j , что $i \neq j$, выполнено $r_i \neq r_j$ или $c_i \neq c_j$.

Сумма по всем n не превосходит 100 000.

Формат выходных данных

Для каждого набора входных данных, выведите одно целое число: минимальное количество действий, которое нужно совершить, чтобы получить хотя бы одну симпатичную фигуру среди данных ладей.

Система оценки

Подзадача	Баллы	Ограничения
1	10	$n \leq 4$
2	10	$n \leq 50$
3	10	$n \leq 200$
4	30	$n \leq 2000$
5	40	$n \leq 10^5$

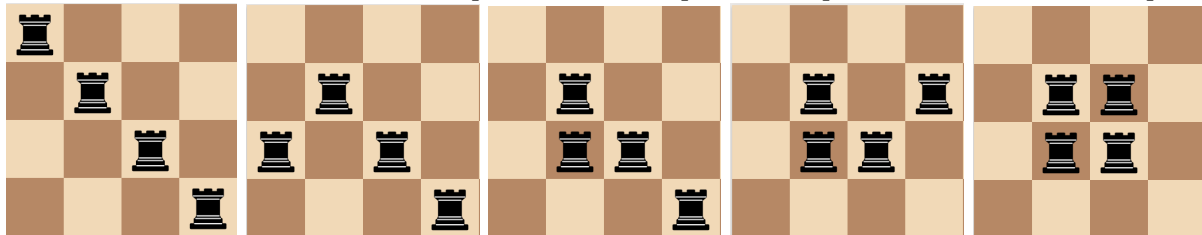


Пример

стандартный ввод	стандартный вывод
5	4
4	2
4 4	1
1 1	3
2 2	3
3 3	
4	
4 4	
4 1	
1 4	
2 2	
6	
3 2	
2 1	
1 2	
3 3	
3 4	
3 1	
5	
1 1	
1 2	
1 3	
1 4	
5 5	
4	
1000000000 1000000000	
1000000000 1	
2 2	
1000000000 999999999	

Замечание

Одно из возможных оптимальных решений для первого набора входных данных из примера:



Одно из возможных оптимальных решений для второго набора входных данных из примера:

