

Задача А. Треугольник минимальной площади

Ограничение по времени: 0,7 с
Ограничение по памяти: 256 МБ

Изрядно поиграв в футбол как футболисты, а после и побыв тренерами, Андрей и Виталий решили придумать свою игру, которая будет очень похожа на футбол.

И они её придумали. Собственно, получился тот же футбол, все правила точно такие же. Но есть одно отличие от классического футбола, привычного нам. В этой игре ворота имеют вид треугольника, и в начале каждой игры команде нужно построить их из металлических палок, которые имеются в наличии.

Возможно, построить ворота можно не единственным способом, поэтому это может влиять на ход игры. Тогда, если действовать наиболее оптимальным способом, стоит сделать такие ворота, чтобы их площадь была наименьшей, чтобы забить было сложнее.

В наличии есть n натуральных различных палок длиной от 1 до k метров. Нужно построить ворота (треугольник) наименьшей возможной площади, то есть нужно выбрать три палки длины a, b, c соответственно таким образом, чтобы:

1) $a < b + c, b < a + c, c < a + b;$

2) из всех возможных троек выбрана такая, что треугольник со сторонами a, b, c будет иметь минимально возможную площадь.

Формат входных данных

В первой строке содержится целое неотрицательное число t — количество тестовых случаев. Первая строка каждого из последующих t тестовых случаев содержит два целых положительных числа n и k — количество палок и ограничение на наибольшую длину палки в тестовом случае.

Вторая строка каждого из тестовых случаев содержит n различных чисел от 1 до k — длины палок.

Гарантируется, что сумма n по всем тестовым случаям не превышает 50 000.

Гарантируется, что сумма k по всем тестовым случаям не превышает 10^5 .

Формат выходных данных

Выведите t строк, среди которых i -я строка содержит ответ на i -й тестовый случай. Если построить треугольник из заданного набора длин нельзя, нужно вывести -1 . Ответ будет считаться правильным, если его относительная или абсолютная погрешность не превосходит 10^{-9} .

Пример

стандартный ввод	стандартный вывод
4	-1
3 3	2.904737509655563
1 2 3	12.968712349342937
4 4	12.968712349342937
1 2 3 4	
3 11	
5 7 11	
6 11	
5 7 8 9 10 11	

Задача В. Перестановка

Ограничение по времени: 1 с

Ограничение по памяти: 256 МБ

В очередной раз во сне к вам пришел дракон... В этот раз его волнуют вопросы, связанные с перестановками.

Каждый вопрос состоит из трёх чисел n , a , b . Необходимо построить такую перестановку длины n , которая удовлетворяет следующим условиям:

- 1) первое число перестановки равно a , последнее — b ;
- 2) любые два соседних числа перестановки таковы, что их НОД (наибольший общий делитель) равен единице;
- 3) количество пар соседних элементов в перестановке, которые различаются больше чем на единицу, должно быть минимально возможным.

Не расстраивайте дракона, ответьте на все его вопросы.

Формат входных данных

Первая строка содержит целое число t — количество вопросов дракона ($1 \leq t \leq 1000$).

Следующие t строк содержат по три целых числа n , a , b — длина перестановки, значения первого и последнего элементов перестановки.

Гарантируется, что сумма всех значений n в тестах не превышает 200 000.

Формат выходных данных

Нужно последовательно вывести ответы для всех t вопросов. Каждый ответ состоит или из одной строки, или из двух. Если невозможно построить ответ для какого-то вопроса, нужно вывести одну строку, содержащую -1 .

Иначе необходимо вывести две строки. Первая содержит одно число — минимальное количество пар соседних элементов в перестановке таких, что они различаются больше чем на единицу. Вторая строка должна содержать построенную перестановку.

Пример

стандартный ввод	стандартный вывод
4	0
5 1 5	1 2 3 4 5
6 4 5	1
7 5 3	4 3 2 1 6 5
4 1 3	2
	5 4 7 6 1 2 3
	-1

Задача С. Командная работа

Ограничение по времени: 3 с

Ограничение по памяти: 256 МБ

Вы работаете в крупной компании. У вас в подчинении находится команда из n людей. Для решения определённой задачи вы можете выбрать непустое множество людей. Стоимость множества из x людей равна x^k .

Найдите сумму стоимостей по всем непустым подмножествам людей.

Формат входных данных

Единственная строка содержит два целых числа n, k — количество людей в подчинении и параметр для подсчёта стоимости ($1 \leq n \leq 10^9, 1 \leq k \leq 5000$).

Формат выходных данных

Выведите ответ на задачу по модулю $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
1 1	1
3 2	24

Задача D. Аэропорты

Ограничение по времени: 2 с

Ограничение по памяти: 256 МБ

Виталий играл 10 лет в футбол за команду «Манчестер Юнайтед». За это время он выиграл много турниров, заработал кучу денег и решил стать бизнесменом. Что-то стукнуло ему в голову, и он решил заняться авиаперелётами. Медленно и планомерно, используя выигранные деньги, Виталий скупал авиакомпании и самолёты и вдруг стал единоличным владельцем всех аэропортов и самолётов в стране.

Андрей играл 10 лет в футбол за команду «Ливерпуль». За это время он выиграл много турниров, заработал кучу денег и решил стать политиком. Медленно и планомерно, используя выигранные деньги и связи, Андрей стал министром по авиаперелётам (внезапно).

У Андрея и Виталия было вечное дерби на футбольном поле, но Виталий был всегда немножко более удачлив. Поэтому Андрей решил каким-нибудь образом срезать доходы Виталия. Он выбрал следующий элегантный способ, чтобы уменьшить количество возможных рейсов.

В стране есть n аэропортов. i -й аэропорт имеет координаты (x_i, y_i) . Андрей очень любит фиксировать константы. В этот раз он зафиксировал константу D и решил, что если манхэттенское расстояние между двумя аэропортами равно по меньшей мере D , то между этими двумя городами обязательно будут проводиться авиаперелёты. В ином случае рейсы функционировать не будут.

Определите наибольшее значение D такое, что для любого аэропорта можно будет добраться в любой другой при помощи авиаперелётов.

Формат входных данных

В первой строке содержится целое число n — количество аэропортов ($2 \leq n \leq 10^5$).

В i -й из следующих n строк содержатся два целых числа x_i, y_i — координаты i -го аэропорта ($0 \leq x_i, y_i \leq 10^9$).

Формат выходных данных

Выведите наибольшее возможное значение константы D , описанной в условии.

Примеры

стандартный ввод	стандартный вывод
6 1 7 8 5 6 3 10 3 5 2 6 10	9
6 9 8 8 8 6 5 3 7 2 6 9 10	8
2 547865075 605997004 539239436 349306506	265316137

Замечание

Напомним, что манхэттенское расстояние между точками (x_1, y_1) и (x_2, y_2) определяется как $|x_1 - x_2| + |y_1 - y_2|$.

Задача E. Универсальные числа

Ограничение по времени: 5 с

Ограничение по памяти: 256 МБ

Мальчик Костя очень любит играть в игры с числами. Вчера папа научил Костю возводить числа в целые неотрицательные степени, после чего задал вопрос: сколькими способами число n может быть представлено как сумма трёх чисел, являющихся степенями одного и того же числа k ?

Другими словами, необходимо определить, сколько существует четвёрок целых неотрицательных чисел (k, a, b, c) при $k > 0$ таких, что верно соотношение $n = k^a + k^b + k^c$. Помогите Косте посчитать это количество.

Формат входных данных

В единственной строке записано целое число n ($4 \leq n \leq 10^{18}$).

Формат выходных данных

Выведите количество подходящих четвёрок.

Примеры

стандартный ввод	стандартный вывод
4	3
5	6
10	9
1234567890	4

Задача F. Красивые массивы

Ограничение по времени: 1 с

Ограничение по памяти: 256 МБ

Красивым называется массив, который удовлетворяет следующим условиям:

- 1) длина массива равна n ;
- 2) каждое число массива — целое число от 0 до $n-1$;
- 3) XOR-сумма всех чисел массива равна 0 (здесь XOR — это побитовое сложение по модулю 2);
- 4) n — степень двойки.

Для заданного n определите, сколько существует красивых массивов длины n .

Два массива различны, если существует такая позиция, в которой их элементы различаются.

Формат входных данных

Единственная строка содержит целое число n ($1 \leq n \leq 10^{18}$).

Формат выходных данных

Выведите ответ на задачу по модулю $10^{18} + 7$.

Примеры

стандартный ввод	стандартный вывод
1	1
2	2
3	0
4	64

Задача G. Карточки

Ограничение по времени: 1 с

Ограничение по памяти: 256 МБ

У мальчика Кости есть n карточек, на которых написано по одному целому числу от 1 до 1000. Костя хочет выбрать какие-то из этих карточек и разложить их парами.

Определите, какое максимальное количество пар карточек может выбрать Костя, чтобы для каждой пары сумма чисел, написанных на них, была равна одному и тому же числу s .

Формат входных данных

В первой строке записано одно целое число n ($2 \leq n \leq 500$).

Во второй строке записаны n целых чисел a_i ($1 \leq a_i \leq 1000$). Числа в строке разделены одиночными пробелами.

Формат выходных данных

Выведите наибольшее количество пар карточек с одинаковой суммой, которые может выбрать Костя.

Примеры

стандартный ввод	стандартный вывод
7 1 2 3 4 5 6 7	3
10 2 6 8 1 3 2 6 6 2 7	4

Задача H. Степень двойки или простое в квадрате?

Ограничение по времени: 1 с

Ограничение по памяти: 256 МБ

Вы очень любите числа. Но больше всего вы любите степени двойки и простые числа. Напомним, что число является *простым*, если оно имеет ровно два *различных* делителя — единицу и само себя. В такой формулировке числа 2, 3, 5, 7 — простые, а числа 100, 33 — составные.

Вас заинтересовало, что больше для заданного k из следующих чисел:

- 1) 2^{k-1} ;
- 2) квадрат k -го простого числа.

Формат входных данных

Единственная строка содержит целое положительное число k . Гарантируется, что k не превышает 100.

Формат выходных данных

Если число 2^{k-1} больше, выведите слово First, иначе выведите слово Second.

Гарантируется, что число k таково, что 2^{k-1} не равно квадрату k -го простого числа.

Примеры

стандартный ввод	стандартный вывод
1	Second
100	First

Задача I. Лыжная гонка

Ограничение по времени: 1 с

Ограничение по памяти: 256 МБ

Вы участвуете в лыжной гонке. Ходят слухи, что главный приз — путёвка на Чемпионат мира по футболу 2022!

Как и каждый другой гонщик, вы должны пройти через n ворот с номерами от 1 до n . Каждые ворота состоят из нескольких эквивалентных контрольных точек, которые можно рассматривать как точки на плоскости с координатами (i, j) для всех целых j между l_i и r_i включительно. Другими словами, i -е ворота — это горизонтальный отрезок с координатами крайних точек (i, l_i) , (i, r_i) , причём в отрезок входят только целочисленные точки. Для прохождения дистанции необходимо пройти через контрольные точки всех ворот в порядке возрастания номера.

К сожалению, лыжник из вас никакой. Вы хотите продумать маршрут, который будет являться прямой линией, проходящей через одну контрольную точку в каждом воротах (по прямой ехать не составляет для вас никакого труда).

Подсчитайте, сколько существует вариантов маршрутов для вас.

Формат входных данных

Первая строка содержит целое число n — количество ворот ($2 \leq n \leq 200\,000$).

Следующие n строк содержат по два целых числа l_i и r_i — описание i -х ворот ($1 \leq l_i \leq r_i \leq 10^9$).

Формат выходных данных

Выведите искомое количество маршрутов.

Пример

стандартный ввод	стандартный вывод
3 1 3 2 3 1 5	6

Задача J. Правильная тренировка

Ограничение по времени: 2 с

Ограничение по памяти: 256 МБ

После тренировочных сборов тренер Виталий понял: чтобы его команда выиграла чемпионат, нужно правильно проводить тренировки.

У Виталия всегда были счёты с Андреем, тренером второй команды города. В этом сезоне команда Андрея начала показывать отменные результаты. А всё дело в том, что футболисты всегда придерживаются определенного правила на тренировках. Пусть в обоих клубах находятся n футболистов. Чтобы провести тренировку правильно, нужно расставить футболистов на поле таким образом, чтобы все манхэттенские расстояния между каждой парой футболистов различались.

Будем считать, что футбольное поле представляет собой квадрат. Левый нижний угол футбольного поля имеет координаты $(-1000, -1000)$, а правый верхний — $(1000, 1000)$.

Футболист может быть расположен в любой точке футбольного поля, которая имеет целочисленные координаты. Таким образом, существует всего $2001 \cdot 2001$ возможных точек для расположения. Отметим, что два футболиста не могут находиться в одной точке.

Помогите Виталию, расставьте n футболистов на футбольном поле для проведения правильной тренировки.

Формат входных данных

Единственная строка содержит целое число n — количество футболистов в команде Виталия ($1 \leq n \leq 43$).

Формат выходных данных

Выведите n строк, в i -й из которых находятся два целых числа x_i, y_i — координаты i -го футболиста. Для любого i должно выполняться условие: $-1000 \leq x_i, y_i \leq 1000$.

Примеры

стандартный ввод	стандартный вывод
1	399 -313
5	399 -313 579 -907 80 738 529 -867 968 -317

Задача K. Игра в дурака

Ограничение по времени: 1 с

Ограничение по памяти: 256 МБ

Самая известная карточная игра на просторах России и Беларуси — игра в «дурака». Опишем правила игры.

В игре используется колода из 36, 52 или 54 карт. Участвуют от двух до шести игроков; в случае игры колодой в 52 и 54 карты может играть до восьми игроков. Старшинство карт в колоде из 36 карт от меньшего достоинства к большему: 6, 7, 8, 9, 10, В, Д, К, Т. Старшинство карт в колоде из 52 и 54 карт: 2, 3, 4, 5, 6, 7, 8, 9, 10, В, Д, К, Т. В колоде из 54 карт есть два

джокера. Старшинство мастей для игры в дурака не определено. Каждому раздаётся по 6 карт, следующая (или последняя, возможно, и любая из колоды) карта открывается, и её масть объявляется козырной для данной игры. Остальная колода кладётся сверху так, чтобы козырная карта была всем видна. Козырный джокер, то есть джокер цвета козырной масти, не отбивается в игре. Цель игры — избавиться от всех карт. Последний игрок, не избавившийся от карт, считается проигравшим (остаётся в «дураках»). Партия в дурака называется кон. Запрещается забирать карты, которые отбили. Отбитые карты идут в отбой (биту).

В ходе игры заходящий игрок кладёт на стол любую из имеющихся у него карт, а отбивающийся игрок (игрок, под которого сделан заход) должен либо побить её, либо взять. Чтобы побить карту, нужно из имеющихся на руках карт положить на неё старшую карту той же масти, либо козыря, если битая карта не козырь. Если битая карта — козырь, то побить её можно только старшим козырем. После того как игрок побил карту, заходящий игрок либо кто-то из поддающих может положить ещё одну карту любой масти, достоинство которой совпадает с достоинством любой из карт, уже лежащих на кону (подкинуть карту). Обычно в роли поддающих могут выступать любые игроки, кроме отбивающегося, но в некоторых вариантах игры есть ограничения, например, только соседними с отбивающимся игроками. Как правило, подкидывать принято по одной карте, причём право подкидывать карту первым принадлежит заходящему, затем, если он отказывается подкидывать дальше, говоря ключевое слово «Бито!», это право переходит к сидящему слева от него и так далее по кругу. Однако иногда допускается возможность подкидывать для любого игрока вне очерёдности и в любом количестве, но так, чтобы общее количество карт, положенных под отбивающегося, не превышало допустимого. Допускать подобные вольности нецелесообразно из-за возможности возникновения конфликтов между поддающими игроками, подкидывающими карты одновременно. Подкидывать можно столько карт, чтобы общее число карт, которые должен отбить отбивающийся, не превышало шести. Однако если у игрока было более 6 карт, а колоды уже нет, то можно кинуть ему на одну карту меньше, чем их у него было изначально. При первом по счёту отбое в игре разрешается подкинуть не более 5 карт. Если отбивающийся игрок не может или не хочет побить хотя бы одну карту, то он обязан принять все карты, которые лежат на столе.

Если же игрок побил все карты (отбился), то все карты, участвовавшие в этом заходе, переворачиваются и складываются в отдельную кучу — отбой, который больше не участвует в игре и подглядывать в который до конца кона нельзя. Пока карты на столе не биты или игрок не взял их себе, брать новые карты из колоды нельзя. После того как сыгран заход, игроки по очереди добирают из оставшейся колоды карты до шести, если, конечно, у них на руках меньше шести карт. Первым берёт недостающее количество карт заходящий игрок, затем карты берут поддающие начиная с игрока, сидящего слева от отбивающегося и так далее по часовой стрелке. Последним карты берёт отбивавшийся игрок. В случае, когда карты в колоде заканчиваются, тот, кто должен был брать, конечно, больше не набирает карт.

Выше были описаны обобщённые правила игры в «дурака». Далее опишем отличия правил классического дурака от правил в текущей задаче.

- 1) Колода карт состоит из 54 карт: 52 обычные, по 13 каждой масти, и два джокера — чёрный и красный. Игроков будет двое.
- 2) Нельзя ходить с джокера. Другими словами, если игра ещё не закончена (у обоих игроков на руках есть карты), то заходящий, который имеет на руках только джокеры, проигрывает.
- 3) Джокером можно бить любую карту при одном условии: цвет карты, которая бьётся, должен совпадать с цветом джокера. После того как джокером была побита какая-нибудь карта, текущий ход сразу же заканчивается и игроки меняются местами: тот, кто отбился, будет ходить следующим.
- 4) Козырь указывается отдельно.

Вы решили поиграть с вашим другом в «дурака». Каждый из вас знает, какие карты находятся на руках у вас и у вашего оппонента. Также вы оба знаете порядок расположения карт в оставшейся колоде. Определите, кто из вас выиграет при оптимальной игре обоих.

Формат входных данных

Первая строка содержит целое число t — количество тестовых случаев ($1 \leq t \leq 10000$).

Каждый тестовый случай начинается с описания карт, которые находятся у вас на руках. Если точнее, первая строка содержит 6 пар символов — карты, которые имеются у вас на руках.

Вторая строка тестового случая содержит описание карт вашего друга в таком же формате, как и в первой строке.

Третья строка тестового случая описывает колоду в том порядке, в котором оставшиеся карты в ней расположены. Первая карта — это верхняя карта в колоде.

Каждая карта описывается парой символов. Первый символ — это старшинство карты, принимающее одно из значений (2, 3, ..., 9; T для 10, J для валета, Q для дамы, K для короля, A для туза). Вторым символом — масть карты, один из четырёх символов: S — пики, C — трефы, D — бубны, H — червы. Пики и трефы носят чёрный цвет, бубны и червы носят красный цвет. Красный джокер описывается как RJ, чёрный джокер — как BJ.

Последняя строка тестового случая содержит один символ — масть козыря в данном тестовом случае.

Рекомендуем изучить примеры к задаче.

Формат выходных данных

Для каждого тестового случая выведите одну строку. Если при оптимальной игре выиграете вы, то необходимо вывести слово Johann, если победит ваш соперник — слово Sebastian, если игра закончится ничьей — слово Bach.

Пример

стандартный ввод	стандартный вывод
2 TC QD 2S TH 4S 3C AS RJ AC 7D 6C BJ 3D 4C 8C AD TD TS 7H JS KD 4H QC 6H 9D 7C 9H JC AH 5H 6S QH KS 5S 5D 3H JD JH 8H QS 2H 4D 5C 9S KH 6D 9C 8D 8S KC 7S 3S 2D 2C S TC 8S JS JD 5C 9C QS 8C 3H 4D 4H 2D QH 7S 7H 3C 2H 7C TD 9H 8D AH 7D QC JH 5D AS 5H 3D JC 2S 6D AC 9D 4C 6S KD 8H 6C 4S RJ KH 3S TS KC KS 5S QD 9S BJ 6H TH AD 2C D	Johann Sebastian

Замечание

Хотя для облегчения чтения в условии в примерах колода записана в нескольких строках, в тестах тестирующей системы все 42 карты записаны через пробел в одной строке.