Задача A. Телесъёмка

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

Для съёмок финального матча «Ротор–Закат» была нанята лучшая съёмочная группа. Несмотря на то, что их работа была выполнена на высочайшем уровне, после просмотра записи выяснилось, что один из игроков Заката ни разу не попал в кадр. Но тренеру интересны действия всех игроков, в том числе и того, которого не оказалось на записи.

Для упрощения задачи будем считать, что игровое поле представляет собой клетчатый прямоугольник $w \times h$ клеток. Каждую секунду игрок обязательно перебегает в соседнюю по стороне клетку. Съёмка же является последовательностью из n кадров, в каждом из которых видно какой-то подпрямоугольник поля с противоположными углами в (x_{i_1}, y_{i_1}) и (x_{i_2}, y_{i_2}). Так как известно, что этот игрок ни разу не попал в кадр, содержимое кадров не важно, важно лишь то, какой участок поля снимался в каждый момент времени.

Ваша задача — помочь тренеру и восстановить какой-либо маршрут, по которому мог перемещаться игрок.

Формат входного файла

В первой строке заданы натуральные числа w и h ($1 \le w, h \le 300$) — ширина и длина поля. Во второй строке задано число n ($1 \le n \le 300$) — число кадров съёмки.

В следующих n строках задано по четыре числа x_{i_1}, y_{i_1}, x_{i_2} и y_{i_2} ($1 \le x_{i_1} \le x_{i_2} \le w$, $1 \le y_{i_1} \le y_{i_2} \le h$) — прямоугольник, соответствующий видимой на i-м кадре части поля.

Формат выходного файла

Выведите n строк, в каждой из которых должно быть по два числа x_i и y_i ($1 \le x_i \le w$, $1 \le y_i \le h$) — координаты клетки, в которой мог оказаться этот игрок на i-й секунде. Выведите «Impossible», если не могло быть такой ситуации, что игрок не попал ни на один из кадров.

Примеры

broadcast.in	broadcast.out
3 2 5 1 1 2 1 2 1 2 1 2 2 3 2 2 2 3 2 2 1 2 2	1 2 2 2 2 1 3 1 3 2

Задача B. Автомобили в Байтландии

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

Ни для кого не секрет, что в Байтландии выпускается очень много различных моделей автомобилей. Хотя со временем устаревшие модели снимают с производства, им на смену приходит все больше новых. Каждая модель автомобиля характеризуется различными параметрами, и один из самых важных — стоимость автомобиля.

Футболисты местного футбольного клуба «Меткий баг» — довольно странные люди, и при покупке автомобиля в первую очередь руководствуются именно стоимостью автомобиля. Они не станут покупать слишком дорогую или дешевую модель. Футболисты считают модель слишком дорогой, если ее стоимость больше средней арифметической стоимости всех моделей, выпускаемых на данный момент, и слишком дешевой, если ее стоимость меньше. Модели со стоимостью, строго равной этому среднему арифметическому, напротив, пользуются большим спросом у футболистов. Компанию, выпускающую автомобили в Байтландии, заинтересовал вопрос: сколько моделей будут пользоваться спросом среди Байтландских футболистов в определенный момент?

Формат входного файла

В первой строке входного файла задано количество запросов n ($1 \le n \le 10^5$). В следущих n строках заданы запросы. Запросы бывают трех видов:

- запрос на добавление модели в производство в виде «$+\ k$», где k — стоимость новой модели ($0 \le k \le 10^{13}$)
- запрос на удаление наиболее устаревшей модели из производства в виде «$-$». Наиболее устаревшей моделью считается та, которая начала выпускаться раньше, чем все остальные, выпускаемые в данный момент. Гарантируется, что при поступлении этого запроса в производстве есть хотя бы одна модель.
- запрос на вывод количества моделей в производстве, стоимость которых равна среднему арифметическому стоимостей всех автомобилей, выпускаемых в данный момент. Запрос поступает в виде строки «?». Гарантируется, что в момент запроса в производстве есть хотя бы одна модель.

Изначально в производстве нет ни одной модели.

Формат выходного файла

На каждый запрос вида «?» выведите ответ в отдельной строке. Ответы выводите в порядке следования запросов во входном файле.

Примеры

cars.in	cars.out
10 + 1 + 4 + 3 + 4 ? - - ? - ?	1 0 1

Задача C. Закат

Имя входного файла:	denisov.in
Имя выходного файла:	denisov.out
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Долгие переговоры руководства ФК «Закат» увенчались успехом: Брюс Бэннер присоединился к команде. Как и положено звезде мирового футбола, он получал немаленькую зарплату.

Однако, одному из прежних игроков команды это не понравилось. Стало ясно, что не избежать волны требований увеличения заработка. После некоторого раздумья, великий тренер Чулано Спагетти нашел способ решить эту проблему.

Всего в команде «Закат» n игроков. Некоторые из них имеют российское гражданство, все остальные — легионеры. К требованиям легионеров тренер относится с осторожностью и предпочтет увеличить им зарплату, чтобы не упустить ценного игрока. А вот российских игроков тренер будет стимулировать снижать их запросы следующим действием: в ближайшем матче не выпускать на поле того игрока с российским гражданством, который потребует себе наибольшую зарплату. Теперь ваша задача определить, кто из команды не выйдет на поле, зная фамилии игроков, их гражданство и зарплату, которую они требуют.

Формат входного файла

В первой строке задано число футболистов n ($1 \le n \le 100$). Далее следуют n строк, содержащих описания футболистов. Каждое описание состоит из перечисленных через пробел зарплаты, требуемой этим игроком, его фамилии и гражданства. Фамилия и гражданство — строки, содержащие только строчные и прописные латинские символы. Длина фамилии и гражданства не превышает 20 символов. У игроков, имеющих российское гражданство, в соответствующем параметре значится «Russia». Зарплата, которую требует игрок — положительное число, не превышающее 10^9. Гарантируется, что в команде есть хотя бы один игрок с российским гражданством.

Формат выходного файла

Выведите фамилию футболиста, который не выйдет на поле. Гарантируется, что такой игрок всегда один.

Примеры

denisov.in	denisov.out
5 5000000 Banner Brazil 1000000 Malafeev Russia 3000000 Denisov Russia 2000000 Kerzhakov Russia 4000000 Criscito Italy	Denisov

Задача D. Футбольные поля

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

Во Флатландии очень любят футбол. Обычные прямоугольные поля давно всем наскучили, так что во Флатландии придумали игру на полях произвольной формы. Форма поля — выпуклый многоугольник. По правилам Ассоциации Флатландского Футбола, каждый стадион должен иметь свою, оригинальную форму, не похожую на форму других стадионов.

Даны два многоугольника, задающих два поля. Вершины каждого многоугольника заданы в порядке обхода по или против часовой стрелки, начиная с произвольной вершины. Возможно, что многоугольники заданы разным порядком обхода. Вам необходимо определить, являются ли они одинаковыми. Два поля называются одинаковыми, если одно из другого можно получить некоторой последовательностью параллельных переносов, отражений относительно любой горизонтальной или вертикальной прямой, а также отражений относительно прямых $y = x$ или $y = -x$.

Формат входного файла

В первой строке входного файла задано число n ($3 \le n \le 10^5$) — количество вершин в многоугольнике. В следующих n строчках находится по два целых числа x и y ($-10^9 \le x, y \le 10^9$) — координаты очередной вершины первого многоугольника.

В следующих n строчках в таком же формате задаются вершины второго многоугольника.

Многоугольники, описаннные во входном файле, выпуклые. Никакие три подряд идущие вершины одного многоугольника не лежат на одной прямой. Вершины каждого многоугольника заданы в порядке обхода по или против часовой стрелки, начиная с произвольной вершины. Возможно, что вершины многоугольников заданы в порядке разных обходов.

Формат выходного файла

В единственной строке выведите слово `YES`, если эти два поля одинаковы, и `NO` иначе.

Примеры

fields.in	fields.out
3 0 0 1 1 1 0 2 2 3 2 3 1	YES

Задача E. Мы - мясо! Мы - газ!

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

Футбольный клуб «Газмяс» в этом сезоне тренирует новый тренер, поскольку методами предыдущего тренера руководство и игроки были слегка не довольны. Новый тренер подходит ко всем вопросам с математической точки зрения. Если раньше перед тренировкой игроки собирались в кружок и слушали тренера, то у нового тренера другой подход. Он строит игроков в линию в алфавитном порядке. А так как тренер окончил Институт Точного Математического Образования, он легко может с точностью до сантиметра определить рост каждого игрока.

Далее задача тренера состоит в том, чтобы определить состав на следующую игру. Для этого он хочет определить сколько у него есть пар сыгранных футболистов. Но у нового тренера понятие «Пара сыгранных футболистов» очень странно. Два футболиста считаются сыгранными, если при построении в алфавитном порядке игрок с большим ростом стоит левее, чем второй футболист, рост обоих футболистов четен, и между ними стоит хотя бы один футболист с нечетным ростом.

Хотя новый тренер и очень умен, но в этом сезоне состав «Газмяса» очень большой (оно и понятно, «Газмяс» собирается выйти в Премьер-лигу). Поэтому он не может сам посчитать количество пар и просит Вас помочь ему. Перед тем, как вы приступили к выполению этого ответственного задания, тренер сообщил Ввм, что рост всех игроков различен.

Формат входного файла

Первая строка содержит одно число N $(1 \le N \le 10^5)$ — количество футболистов. Вторая строка содержит N чисел, идущих через пробел — рост футболистов. Рост каждого футболиста положителен и не превышает N. Рост всех футболистов различен. Футболисты уже упорядочены по алфавиту.

Формат выходного файла

Выведите одно число — количество пар сыгранных футболистов.

Примеры

gazmyas.in	gazmyas.out
5 4 3 2 5 1	1
5 4 2 3 5 1	0

Задача F. Метро

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

Однажды, в 2111 году, на съезде межгалактической футбольной федерации единогласно решили доверить право на проведение в 2118 году чемпионата вселенной по футболу Берляндскому футбольному союзу. И вот на долю мэра Берляндска пала нелегкая доля по организации такого масштабного события у себя в городе. И главная проблема, которая перед ним встала, заключается не в постройке стадионов, а в подготовке инфраструктуры: транспорта, гостиниц, системы общественного питания. Вам же необходимо помочь ему с решением проблемы постройки метро в Берляндске.

Болельщики из определенной галактики могут ездить только на удобном для них виде составов метро. Для того, чтобы гости остались довольны, необходимо сделать на перегонах между станциями метро специальные составы. У каждой галактики есть свой уникальный тип состава, ездящий по своим уникальным рельсам и только по ним.

Про каждый перегон между двумя станциями метро известно, какого он типа (болельщики из какой галактики должны по нему ездить). Было решено, что будет куплено минимальное возможное количество составов метро, при котором через каждую станцию будут курсировать составы всех типов, для которых из этой станции есть подходящий перегон. Если состав курсирует через некоторую станцию метро A, то он курсирует также через все станции, до которых можно добраться из A по перегонам подходящего типа. Вам необходимо определить, сколько необходимо купить составов каждого типа.

Формат входного файла

В первой строке входного файла даны три целых числа n, m и k ($1 \le n \le 10^5$, $1 \le m \le 2 \cdot 10^5$, $1 \le k \le 10^5$) — количество станций метро, количество перегонов между ними и количество типов составов.

В следующих m строках описаны перегоны, по одному в строке. Каждый перегон описан тремя числами: номера станций метро, которые соединяет этот туннель, и номер типа состава, который может проехать по этому перегону. Номера станций — положительные числа, не превышающие n. Тип состава — положительное число, не превышающее m.

Формат выходного файла

Выходной файл должен состоять из k строк.

В i-ой строке выведите количество составов типа i, которое необходимо купить.

Примеры

metro.in	metro.out
4 3 2 1 2 1 2 3 2 3 4 1	2 1

Задача G. Палиндромы

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

Василий после тяжёлого футбольного мятча любит отдыхать, а именно смотреть кино или играть в компьютерные игры. Но после очередного матча ему надоело играть в футбольные симуляторы и смотреть сериалы. Тогда он решил взять у своего брата-близнеца игру `Palin`.

Игра заключалась в следующем: дана строка S и k правил замены одного символа на другой. То есть, если i -й символ строки будет равен a_j, то его можно будет заменить на символ b_j, где a_j и b_j — символы j-го правила замены. Кроме того, если символ номер i в строке уже был изменен, то изменить его еще раз невозможно. Требуется из данной строки получить палиндром. Напомним, что палиндромом является строка, которая одинаково читается и слева направо, и справа налево.

И тут Василий задался следующим вопросом: за какое минимальное количество операций замены такое возможно?

Формат входного файла

В первой строке входного файла дана строка S $(0 < |S| \le 10^5)$, содержащая только маленькие латинские буквы. Во второй строке дано число k $(0 \le k \le 26)$ — количество правил. Следующие k строк содержат по два символа a_i и b_i (маленькие латинские буквы) — символы в очередном правиле замены. Все a_i различны.

Формат выходного файла

Если из данной строки можно получить палиндром, следуя описанным правилам, то в первой строке выходного файла выведите минимальное количество операций замены, выполнив которые это можно сделать. Во второй строке в таком случае выведите номера символов, которые необходимо изменить. Символы в строке нумеруются с единицы.

Если же получить из данной строки палиндром невозможно, выведите в первой строке выходного файла -1.

Примеры

palindromes.in	palindromes.out
abc 1 a c	1 1
abc 1 b c	-1
abc 2 c d d a	-1

Задача H. Стадион

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

В Берляндии скоро пройдет чемпионат мира по футболу, и сейчас активно ведется его подготовка. Для проведения особо важных матчей и церемоний открытия и закрытия планируется построить новый стадион, который должен стать самым большим из известных человечеству. Однако, если стадион будет очень большим, то зрители могут рассаживаться очень долго. Для решения этой проблемы был создан специальный отдел оптимизации стадиона.

По проекту, зрительские места на стадионе будет разбиты на несколько одинаковых секторов. Сектор представляет из себя нескольких рядов кресел по m кресел в каждом ряду с двумя проходами по бокам, так что к каждому месту можно будет пройти справа или слева. В каждом ряду кресла пронумерованы слева направо. Способом рассадки зрителей в одном ряду отдел называет перестановку чисел от единицы до m, соответствующую порядку, в котором зрители занимают свои места в этом ряду. Хорошим способом назовем такой способ, при котором никакому зрителю не придется проходить на пути к своему месту мимо уже сидящего человека. Вас же попросили посчитать, сколько существует различных таких способов. Считается, что очередной зритель начинает идти к своему месту только тогда, когда предыдущий за ним (если такой был) уже занял свое место.

Формат входного файла

В входном файле содержится одно целое число m ($1 \le m \le 10^{18}$) — число мест в каждом ряду.

Формат выходного файла

В выходной файл выведите одно число — число различных хороших способов рассадки зрителей по модулю $10^9 + 7$.

stadium.in	stadium.out
1	1
3	4