

Problem A. Another Hat Problem

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 512 megabytes

Игра в Шляпу — очень популярный вид досуга в ЛКШ. Недавно для характеристики игрока предложили использовать параметр ъуъ .

Про параметр известно, что его значение задаётся вещественным числом.

Оказалось, что два игрока являются идеально совместимыми в следующем случае. Пусть у одного из них ъуъ равен a , а у другого ъуъ равен b . Тогда два игрока идеально совместимы в случае, если все числа $a + b$, $a \cdot b$, $a - b$, $b - a$, a/b , b/a являются целыми.

В качестве примера к подготовленной статье аналитик хочет вычислить количество различных упорядоченных пар (a, b) таких, что $l \leq a, b \leq r$ и игроки с соответствующими значениями идеально совместимы.

Input

Входные данные содержат два целых числа l и r — границы параметра ъуъ ($-10^9 \leq l \leq r \leq 10^9$).

Output

Выведите одно целое число — количество упорядоченных пар (a, b) , определяющих идеально совместимых игроков и удовлетворяющих неравенству $l \leq a, b \leq r$, или -1 , если таких пар бесконечно много. Заметим, что, например, пары $(1, 2)$ и $(2, 1)$ считаются различными.

Example

standard input	standard output
0	1
1	

Problem B. Баскетбольные мячи

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 512 megabytes

Руководство Летней Киберспортивной Школы решило тренировать школьников и классическому спорту тоже. Для этой цели взяли нового спортивного тренера Демида.

Однако его педантичность оказалась сюрпризом даже для руководства школы: увидев несколько мячей, слегка отличающихся радиусом, Демида заявил, что отказывается проводить тренировку разными мячами. В итоге Вас, как опытного программиста, наняли для того, чтобы выбрать необходимое количество мячей.

Внешне все мячи почти одинаковы, различаются они только радиусами. С помощью специального устройства вы определили по четыре точки с целыми координатами на поверхности каждого мяча. Теперь ваша задача — отобрать наибольшее количество мячей с одинаковым радиусом, чтобы новый спортивный тренер смог провести тренировку.

Input

В первой строке задано одно целое число N ($1 \leq N \leq 20$) — количество мячей, имеющих в распоряжении Школы. Каждая из последующих N строк содержит координаты четырёх точек на поверхности каждого из N мячей в порядке $x_1 y_1 z_1 x_2 y_2 z_2 x_3 y_3 z_3 x_4 y_4 z_4$ (то есть сначала идут координаты первой точки, затем второй, затем третьей и затем четвёртой). Все $x_i y_i z_i t_i$ — целые, не меньше нуля и не больше 170. Гарантируется, что тетраэдр с заданными четырьмя точками в качестве вершин имеет ненулевой объём.

Output

Выведите одно целое число — наибольшее число мячей с одинаковым радиусом, которые команда может предоставить Демиду для проведения тренировки.

Example

standard input	standard output
4	2
1 0 1 2 1 1 1 1 2 1 2 1	
3 2 2 2 2 3 2 3 2 2 1 2	
6 4 4 4 4 6 4 6 4 4 2 4	
2 2 1 1 2 2 2 1 2 0 1 2	

Problem C. Восстановление последовательности

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 256 megabytes

Для последовательности целых чисел a_1, a_2, \dots, a_n определим её схему монотонности как последовательность s_1, s_2, \dots, s_{n-1} , состоящую из символов $<$, $>$ и $=$. Символ s_i строится как знак отношения между a_i и a_{i+1} . Например, для последовательности $2, 4, 3, 3, 5, 3$ схема монотонности выглядит как $<, >, =, <, >$.

Считается, что последовательность b_1, b_2, \dots, b_{n+1} со схемой монотонности s_1, s_2, \dots, s_n , реализует другую схему монотонности s'_1, s'_2, \dots, s'_k , если для каждого $i = 1, 2, \dots, n$ выполняется $s_i = s'_{((i-1) \bmod k)+1}$.

Иначе говоря, последовательность s_1, s_2, \dots, s_n может быть получена путём повторения последовательности s'_1, s'_2, \dots, s'_k и удаления соответствующего суффикса из результата. Например, последовательность $2, 4, 3, 3, 5, 3$ реализует, наряду со многими другими, следующие схемы:

- $<, >, =$
- $<, >, =, <, >$
- $<, >, =, <, >, <, <, =$
- $<, >, =, <, >, =, >, >$

Вам заданы целочисленная последовательность a_1, a_2, \dots, a_n и схема монотонности s_1, s_2, \dots, s_k . Ваша задача — найти наибольшую подпоследовательность $a_{i_1}, a_{i_2}, \dots, a_{i_m}$ ($1 \leq i_1 < i_2 < \dots < i_m \leq n$) заданной последовательности, которая реализует заданную схему.

Input

В первой строке входного файла заданы два целых числа n и k ($1 \leq n \leq 500\,000$, $1 \leq k \leq 500\,000$) — длина последовательности a и схемы монотонности s соответственно.

Во второй строке входного файла задана последовательность a , то есть n целых чисел a_i ($1 \leq a_i \leq 1\,000\,000$).

Третья строка содержит схему монотонности s , то есть k символов s_j (каждый символ может быть или $<$, или $>$, или $=$).

Output

В первой строке выходного файла выведите целое число m — наибольшая длина подпоследовательности a_1, a_2, \dots, a_n , реализующей заданную схему s_1, s_2, \dots, s_k .

Во второй строке выведите m чисел $a_{i_1}, a_{i_2}, \dots, a_{i_m}$ — соответствующую подпоследовательность. Если таких подпоследовательностей несколько, выведите любую из них.

Example

standard input	standard output
7 3	6
2 4 3 1 3 5 3	2 4 3 3 5 3
< > =	

Problem D. Группировка по параллелям

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 512 megabytes

В Летней Китайской Школе действует система социального рейтинга. В лагере находятся N школьников, рейтинг школьника — целое неотрицательное число.

Сенсэй Сергееч хочет разделить их на L параллелей, где L — делитель N . Каждая параллель состоит из школьников, чьи номера дают одинаковые остатки при делении на L .

Сколькими способами можно выбрать два целых числа i и j ($1 \leq i < j \leq N$) таким образом, что после того, как мы поменяем местами школьников с номерами i и j , суммарный социальный рейтинг учеников в каждой из L параллелей стал одинаковым?

Input

Первая строка входных данных содержит два целых числа N и L ($1 \leq N \leq 2 \cdot 10^5$, $1 \leq L \leq N$, N делится на L).

Вторая строка содержит N целых чисел. i -е из этих чисел A_i задаёт социальный рейтинг школьника с номером i ($0 \leq A_i \leq 2 \cdot 10^6$).

Output

Выведите одно целое число — количество способов поменять местами ровно двух школьников так, чтобы сумма рейтингов в каждой из L параллелей стала одинаковой.

Examples

standard input	standard output
4 2 1 5 3 3	2
6 2 1 2 3 4 5 6	0

Problem E. Дешифровка

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 512 megabytes

Это интерактивная задача

В Летнем Клубе Шулеров для жеребьёвки перед турниром по покер-сету применяется следующая система.

На вход подаётся битовая строка длины N . Внутри системы записаны K чисел от 1 до N — номера битов (начиная с единицы). Система считает количество единичек среди этих K битов в строке запроса. Если оно чётно, выдаётся 0, если нечётно — выдаётся 1.

Информация о системе тщательно охраняется. Известны только длина битовой строки N и количество используемых при генерации битов K . Чтобы взломать эту систему и использовать ради высших целей, вы решили выяснить номера битов.

Для этого у вас есть не более 200 запросов до того, как система заметит подозрительные действия и запретит вам доступ. Запрос имеет вид $? S$, где S — битовая строка длины N .

Если вы готовы назвать номера битов, выведите ответ в виде $! x_1 \dots x_K$, где x_i — номера битов (биты занумерованы с единицы, начиная с самого младшего). Это действие запросом не считается.

Interaction Protocol

Взаимодействие начинает программа жюри, выводя два целых числа N и K ($1 \leq N \leq 10^4$, $1 \leq K \leq \min(10, N)$) — длину битовой строки и количество используемых системой бит.

Затем вы задаёте запросы в формате $? S$, где S — битовая строка длины N . Программа жюри отвечает 0 или 1 в соответствии с условиями задачи.

Если вы готовы вывести ответ, выведите $! x_1 \dots x_K$. Номера позиций могут быть указаны в произвольном порядке.

Не забывайте после каждого запроса или вывода ответа сбрасывать буфер ввода-вывода командой `flush`.

Гарантируется, что интерактор не является адаптивным, то есть в каждом тесте набор используемых битов определён заранее и не меняется за время взаимодействия.

Example

standard input	standard output
3 1	
0	? 100
1	? 101
	! 1

Problem F. Есть сигнал!

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 512 megabytes

Сеть кабельного интернета в Летней Компьютерной Школе состоит из n узлов и m **односторонних** соединений. Сигнал передаётся на некоторые узлы (называемые передающими узлами). Остальные узлы могут принимать сигнал, если существует (не обязательно прямое) соединение, ведущее из передающего узла в данный узел.

Передающие узлы выбираются так, чтобы, во-первых, обеспечить целостность сети (то есть чтобы все узлы сети могли принимать сигнал), и, во-вторых, количество передающих узлов было минимально.

Однажды в лагере появился Егор Антонов, который каждый день приводит в негодность по одному соединению. Поэтому для сохранения целостности сети каждый день набор передающих узлов выбирается заново. Требуется написать программу, которая вычисляла бы наименьшее возможное количество передающих узлов для каждого из дней после появления Егора.

Input

В первой строке входных данных заданы два целых числа n и m ($1 \leq n, m \leq 100\,000$) — соответственно количество узлов в кабельной сети и количество соединений. В следующих m строках описываются соединения: каждая из этих строк содержит два целых числа a и b ($1 \leq a, b \leq n, a \neq b$) и описывает однонаправленное соединение из узла a в узел b . Между любыми двумя узлами в одном направлении существует не более одного соединения. Следующая строка содержит целое число k ($1 \leq k \leq m$) — количество соединений, которые будут повреждены. Далее следуют k строк. i -я из них содержит номер (в порядке перечисления во входном файле, начиная с 1) соединения, повреждённого в i -й день. При этом все повреждённые соединения попарно различны.

Output

Выведите k строк, i -я из которых содержит минимальное количество передающих узлов, необходимых для обеспечения целостности сети на i -й день после появления Егора Антонова.

Example

standard input	standard output
3 2	2
1 2	3
2 3	
2	
2	
1	

Problem G. Ёлки и экология

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 512 megabytes

Территорию леса можно представить как бесконечную матрицу, строки и столбцы в которой пронумерованы целыми положительными числами. Давным-давно в квадрате $(1, 1)$ выросла первая ёлка. Затем каждый год в каком-то одном квадрате (x, y) , в котором уже росла хотя бы одна ёлка, количество ёлок уменьшалось на 1, но зато в квадратах $(x, y + 1)$ и $(x + 1, y)$ появлялось по новой ёлке.

При постройке новой базы отдыха для ЛКШ, расположенной посреди леса, весь этот лес был вырублен. Но после окончания строительства, владельцы базы решили возместить ущерб экологии и попробовать восстановить лес...

При восстановлении леса ёлки сажают согласно утверждённому плану посадки: N деревьев одно за другим, для каждой ёлки указан квадрат, в который её надо посадить. В первый же момент времени, когда конфигурация ёлок будет полностью совпадать с возможной конфигурацией ёлок на какой-то год развития леса, экологи остановят процесс и дадут лесу развиваться естественно.

По заданному плану восстановления леса требуется определить, после какого дерева экологи остановят процесс высадки, и произойдёт ли это вообще. Если процесс высадки будет остановлен, выведите список квадратов, в которых количество ёлок уменьшалось при соответствующем процессе роста леса.

Input

Первая строка входных данных содержит одно целое число N ($1 \leq N \leq 5 \cdot 10^5$) — количество ёлок, которые высаживаются согласно плану. Каждая из последующих N строк содержит два целых числа x_i и y_i — координаты квадрата, в который высаживается очередная ёлка ($1 \leq x_i, y_i \leq 5 \cdot 10^5$).

Output

Если ни в какой момент времени (включая последнее высаженное дерево) экологи не остановят процесс высадки, выведите -1. Иначе выведите количество деревьев T , которое будет высажено, и последовательность естественного развития леса, которая могла бы привести к этому состоянию леса естественным путём. Последовательность должна состоять из $T - 1$ строк и на каждом шаге последовательно должны быть указаны квадраты, в которых количество ёлок уменьшалось.

Examples

standard input	standard output
4 2 2 1 2 3 1 2 3	3 1 1 2 1
4 4 1 3 2 2 3 1 4	-1

Problem H. Жёсткие правила спортивной мафии

Input file: standard input
Output file: standard output
Time limit: 3 seconds
Memory limit: 512 megabytes

В спортивной мафии возможные спорные ситуации за игровым столом регулируются в соответствии с Кодексом. Кодекс представляет собой сложный документ, в котором на каждой странице встречаются ссылки на другие страницы...

В редакции журнала «Мафия для чайников» решили выпустить новое издание кодекса.

Издание представляет собой книгу из N страниц, обладающую следующими свойствами:

- На каждой странице указаны ссылки на одну или несколько страниц. Все ссылки идут на страницы с номерами, большими, чем текущая.
- С первой страницы можно по цепочке ссылок попасть на любую страницу.
- С любой страницы можно по цепочке ссылок попасть на N -ю страницу.

Сколько существует способов расставить таким образом ссылки на страницах книги? Так как ответ может быть очень большим, выведите остаток от его деления на 998 244 353.

Input

Входные данные содержат одно целое число N ($2 \leq N \leq 500$) — количество страниц в издании.

Output

Выведите одно целое число — остаток от деления количества способов расставить ссылки в N -страничной книге на 998 244 353.

Examples

standard input	standard output
3	2
5	122

Problem I. Забег

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 512 megabytes

Настало время очередного антифотокросса от Стаса, который проводится ежегодно. Правила этого антифотокросса несколько отличаются от привычных вам:

Даны N контрольных точек; все эти точки имеют целые координаты на плоскости.

Участник выбирает маршрут со стартом в одной из этих точек и проходящий через другие точки таким образом, что расстояния между двумя соседними контрольными точками **строго возрастают**. Участник всегда бежит по самой короткой дистанции между двумя контрольными точками; при этом имеется в виду обычное евклидово расстояние, то есть $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$.

Участники получают один бонусный балл за посещение каждой контрольной точки. При этом разрешено посещать одну и ту же контрольную точку несколько раз — каждое посещение даст бонусный балл. Единственное ограничение — нельзя посещать одну и ту же контрольную точку два раза подряд.

Найдите максимальное количество бонусных баллов, которые участник сможет набрать, выбирая стартовую контрольную точку и маршрут оптимальным образом.

Input

Первая строка входных данных содержит одно целое число N ($2 \leq N \leq 2000$) — количество контрольных точек. Каждая из последующих N строк содержит по два целых числа x_i, y_i ($0 \leq x_i, y_i \leq 10^9$) — координаты очередной контрольной точки. Гарантируется, что никакие две контрольные точки не совпадают.

Output

Выведите одно целое число — максимальное количество бонусных баллов, которое может собрать участник антифотокросса.

Examples

standard input	standard output
4 0 0 1 0 0 2 2 3	6
2 4 5 1 7	2

Problem J. Исправляем несправедливость

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 512 megabytes

В турнире по настольным играм принимают участие N школьников. Соревнование состоит из N поединков. В каждом поединке участвуют ровно два школьника, поединок заканчивается победой одного из участников. В течение соревнования некоторые школьники могут участвовать в нескольких поединках, некоторые — ни в одном. В i -м поединке принимают участие школьники a_i и b_i .

Преподаватели считают соревнование *справедливым*, если каждый участник соревнования победил кого-то как минимум один раз.

По заданному расписанию поединков определите количество различных вариантов исходов поединков, при котором соревнование будет названо *справедливым*. Так как ответ может быть очень большим, выведите его остаток от деления на 998 244 353. Два варианта считаются разными, если исход хотя бы одного поединка различается.

Input

Первая строка входных данных содержит одно целое число T — количество тестовых примеров ($1 \leq T \leq 2 \cdot 10^5$).

Далее следуют тестовые примеры. Первая строка каждого тестового примера содержит одно целое число N ($3 \leq N \leq 2 \cdot 10^5$) — количество поединков и участников в соревновании. Каждая из последующих двух строк содержит по два различных целых числа a_i и b_i ($1 \leq a_i, b_i \leq N$) — номера двух школьников, встречающихся в поединке. Ни одна пара школьников не встречается в одном и том же соревновании дважды.

Гарантируется, что сумма N для всех тестовых примеров не превосходит $2 \cdot 10^5$.

Output

Для каждого тестового примера выведите одно целое число — остаток от деления количества различных вариантов исходов поединков, при котором соревнование будет справедливым, на 998 244 353.

Example

standard input	standard output
2	2
5	0
1 2	
2 3	
1 3	
4 1	
5 4	
5	
1 2	
2 3	
3 4	
4 1	
1 3	

Problem K. Йога

Input file: standard input
Output file: standard output
Time limit: 5 seconds
Memory limit: 512 megabytes

В Летней Кулинарной Школе ввели новый тип альтернативной зарядки — йогу. Она стала настолько популярной, что места на неё ученики занимают заранее. Всего есть N учеников. По графику тренировок i -й ученик начинает тренировку в момент времени a_i , и должен тренироваться как минимум до момента времени b_i .

При этом Демид установил в группе следующее правило: если ученик X начал тренироваться **позже**, чем ученик Y , то X не может прекратить тренировку раньше, чем Y , чтобы не демотивировать того (то есть X должен тренироваться дополнительно, пока Y не закончит тренировку, даже если по своему графику X уже может закончить).

Вам даны графики тренировок всех учеников в ЛКШ. Для каждого из них выясните, в какой самый ранний момент этот ученик может закончить тренировку.

Input

Первая строка входных данных содержит одно целое число N — количество учеников ($1 \leq N \leq 2 \cdot 10^5$). i -я из последующих N строк содержит два целых числа a_i и b_i — время начала и конца тренировки для i -го ученика ($1 \leq a_i < b_i \leq 10^9$). Гарантируется, что все $2N$ чисел a_i и b_i попарно различны.

Output

Для каждого ученика выведите в новой строке одно целое число — время, когда он фактически закончит тренировку с учётом собственного графика и правила, установленного Демидом.

Examples

standard input	standard output
3	6
2 6	4
1 4	6
3 5	
3	6
5 6	4
3 4	2
1 2	

Problem L. Командная работа

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 512 megabytes

Летом 2023 года ЛКШ проводит августовскую смену на новой базе. Двенадцатой параллели выпало заниматься на огромном поле в виде выпуклого n -угольника. Внутри этой «аудитории» есть ровно k посадочных мест. По счастливому стечению обстоятельств в параллели двенадцать ровно k учеников.

8 августа преподаватели параллели двенадцать решили провести эксперимент. Они хотят разбить поле на $n - 2$ части, в каждой из которых проведут лекцию по определенной теме. По техническим причинам, разделить поле можно только с использованием $n - 3$ ограждений, представляющих из себя отрезки из одной вершины многоугольника в другую. Ограждения, конечно же, не могут пересекаться или проходить через посадочное место (иначе какому-то школьнику будет негде сидеть).

Поскольку преподаватели хотят проверить способность школьников к командной работе, во время каждой лекции школьники разобьются на пары, где один школьник будет смотреть на доску, а другой будет слушать лектора, и после лекции каждая пара вместе пройдет тест по теме занятия. Ввиду этого, в каждой образовавшейся секции должно быть четное число школьников.

Вас просят вычислить количество способов разбить поле, чтобы удовлетворить всем требованиям преподавателей. Поскольку ответ может быть очень большим, выведите его по модулю m .

Input

Первая строка входного файла содержит три целых числа n , k и m ($4 \leq n \leq 600$, $2 \leq k \leq 20\,000$, $2 \leq m \leq 20\,000$, k четно) — количество вершин многоугольника, образующего поле, количество школьников и модуль m , который потребуется для вывода ответа.

Каждая из последующих n строк содержит два целых числа x_i и y_i ($-15\,000 \leq x_i, y_i \leq 15\,000$) — координаты i -й вершины поля. Вершины заданы в порядке обхода по часовой стрелке.

Каждая из последующих k строк содержит два целых числа p_j , q_j ($-15\,000 \leq p_j, q_j \leq 15\,000$) — координаты j -го посадочного места.

Output

Выведите одно целое число — остаток от деления количества описанных в условии разбиений на m .

Example

standard input	standard output
5 4 10 5 5 3 0 -1 -1 -3 4 1 10 1 0 -1 0 1 6 -2 5	3

Problem M. Ленивые культорги

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 512 megabytes

Для потенциальных участников вечернего мероприятия культоргам требуется сгенерировать номера команд. В качестве номеров могут использоваться любые целые числа от -10^9 до 10^9 , однако организаторы хотят, чтобы соблюдались следующие правила:

- Сумма всех номеров должна быть равна заданному целому числу X .
- Из-за особенностей компьютера на котором проходит регистрация команд произведение попарных разностей всех сгенерированных номеров не должно делиться на заданное простое число p .
- Количество сгенерированных номеров должно быть максимально при условии соблюдения двух первых требований.

Поскольку у культоргов сейчас много других дел, они попросили вас сгенерировать любой подходящий набор номеров.

Input

Первая строка входных данных содержит два целых числа, описанных в условии, X и p ($-10^9 \leq X \leq 10^9$, $2 \leq p \leq 10^5$, p — простое)

Output

В первой строке выведите одно целое число M — максимальное количество сгенерированных номеров. Во второй выведите M целых чисел, не превосходящих 10^9 по абсолютной величине, сумма которых равна X , а произведение всех $M(M - 1)/2$ попарных разностей не делится на p .

Examples

standard input	standard output
3 2	2 5 -2
4 2	1 4