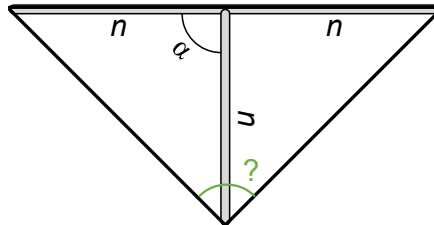


Задача А. Изгороди

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Фермер Джон прокопал прямолинейную канаву длиной $2 \cdot n$ метров. Затем от середины канавы прокопал под углом α к предыдущей короткую канаву длиной n метров, после чего соединил концы канав изгородями.

Найдите величину угла между изгородями у конца короткой канавы.



Формат входных данных

Первая строка содержит одно целое число n ($1 \leq n \leq 1000$). Вторая строка содержит целое число α — величину угла между канавами в градусах ($0 < \alpha < 180$).

Формат выходных данных

Выведите одно целое число — величину угла у конца короткой канавы в градусах, округлённую до ближайшего целого числа.

Пример

стандартный ввод	стандартный вывод
2 90	90

Задача В. Анализ крипторынка

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

На криптобирже появились три новых вида монет. Начальная цена каждой монеты — 1 рубль. Монеты стали пользоваться популярностью, и их цены стали расти: у монет первого вида в a_1 раз, второго — в a_2 раз, а третьего — в a_3 раз за сутки, где a_i — целые положительные числа. Повышение котировок происходит дискретно раз в сутки.

В момент начала продаж Алиса купила по одной монете первого и второго видов, а Боб — одну монету третьего вида. Требуется найти, через сколько дней суммарная стоимость монет Алисы будет в точности равна стоимости монет Боба, или определить, что такого момента не будет.

Формат входных данных

Первая строка содержит целое число a_1 , вторая — целое число a_2 , а третья — целое число a_3 ($1 \leq a_i \leq 10^9$).

Формат выходных данных

Если ни в какой момент в будущем суммарная стоимость монет Алисы не будет равна стоимости монеты Боба, выведите -1 . Иначе выведите номер первого дня, в который будет достигнуто равенство.

Примеры

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

Задача С. Взять след!

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Следом квадратной матрицы B_{ij} называется сумма элементов B_{ii} , расположенных на главной диагонали.

Дана последовательность целых чисел a_i . Требуется расставить числа из последовательности в *непустую* квадратную матрицу B_{ij} так, чтобы её след был максимально возможным. При этом, если число x присутствует в последовательности a_i ровно k раз, то в матрице B_{ij} оно должно присутствовать не более k раз.

Формат входных данных

Первая строка входных данных содержит одно целое число n — длину последовательности a_i ($1 \leq n \leq 10^5$).

Последующие n строк содержат по одному целому числу каждая, i -я из них содержит a_i — i -й элемент последовательности a ($-10^9 \leq a_i \leq 10^9$).

Формат выходных данных

Выведите одно целое число — максимально возможное значение следа матрицы B .

Пример

стандартный ввод	стандартный вывод
6	2052
31	
10	
2021	
-11	
0	
0	

Задача D. Градусы, радианы, градусы

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

На уроке математики Алиса загадала угол, измеряемый целым числом n градусов (от 0 до 359) и передала Бобу запись величины угла в одной из трёх популярных единиц — градусах, радианах или *градах* (метрическая единица измерения углов; прямой угол содержит 100 град). Запись содержит ровно 6 знаков после десятичной точки (и отличается от точного значения менее, чем на $5 \cdot 10^{-7}$).

Боб попросил вас помочь с восстановлением значения n , а также выяснить, какие единицы Алиса использовала для записи.

Формат входных данных

Первая строка входных данных содержит одно целое число T — число тестовых примеров ($1 \leq T \leq 20$). Каждая из последующих T строк содержит одно вещественное число, записанное ровно с 6 знаками после десятичной точки — результат вычислений Алисы.

Гарантируется, что числа были получены описанным в условии задачи способом.

Формат выходных данных

Если однозначно восстановить значение n и использованные единицы измерения невозможно, выведите -1 . Иначе выведите два целых числа — исходное значение n и тип единицы измерения: 1 для градусов, 2 для радианов и 3 для градов.

Пример

стандартный ввод	стандартный вывод
3	-1
0.000000	180 2
3.141593	1 3
1.111111	

Замечание

В первом примере Боб может восстановить $n = 0$, но получить какую-либо информацию о единицах измерения не получится, поэтому ответ -1 .

Задача Е. Подстроки и подпоследовательности

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Рассмотрим строку T , составленную из строчных букв английского алфавита, и построим два множества: множество S_1 всех различных подстрок строки T и множество S_2 всех различных подпоследовательностей строки T .

Например, для строки «*icrc*» S_1 состоит из пустой строки, «*i*», «*c*», «*r*», «*ic*», «*cr*», «*rc*», «*icr*», «*rcs*» и «*icrc*». В S_2 , помимо этих строк, входят строки «*ir*», «*cc*», «*irc*» и «*iccc*».

Назовём строку *необычной*, если $S_1 = S_2$. Отсортируем все необычные строки по возрастанию длины, а строки равной длины — в лексикографическом порядке. Ваша задача — найти n -ю необычную строку.

Формат входных данных

Входные данные содержат одно целое число n ($1 \leq n \leq 10^6$).

Формат выходных данных

Выведите n -ю в соответствии с описанным в задаче упорядочением необычную строку.

Примеры

стандартный ввод	стандартный вывод
1	a
27	aa

Задача F. Если сложить...

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Дана последовательность a_i , состоящая из n целых положительных чисел, больших единицы. Требуется построить такую последовательность из n положительных рациональных чисел b_i , что сумма $S = \sum_{i=1}^n a_i^{b_i}$ является наименьшим возможным целым числом.

Формат входных данных

Первая строка входных данных содержит целое число n ($1 \leq n \leq 10^4$). Вторая строка входных данных содержит n целых чисел a_i ($2 \leq a_i \leq 10^9$).

Формат выходных данных

Выведите одно целое число — наименьшее возможное целое значение S .

Пример

стандартный ввод	стандартный вывод
5 2 3 4 5 6	18

Задача G. Генерация ключей

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

В криптографической системе «EasyCrypt» ключом может быть любое неотрицательное целое число, двоичная запись которого содержит ровно K единиц, и не превосходящее заданного целого положительного числа N .

Вычислите, сколько различных ключей существует для заданных N и K . Так как ответ может быть очень большим, выведите остаток от его деления на простое число 998 244 353.

Формат входных данных

Первая строка входных данных содержит одно целое число N , записанное в *шестнадцатеричной* системе счисления без ведущих нулей ($1 \leq N < 16^{250}$). Цифры, большие 9, обозначаются заглавными латинскими буквами от 'A' до 'F'.

Вторая строка содержит одно целое число K — число бит в ключе, равных единице ($0 \leq K \leq 1000$).

Формат выходных данных

Выведите одно число — остаток от деления количества различных ключей, существующих для заданных N и K , на простое число 998 244 353.

Примеры

стандартный ввод	стандартный вывод
1F 2	10
31 3	17

Задача Н. Супердевятка

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

В чемпионатах по спортивной «Своей игре» часто используется схема проведения финала, называемая «супердевятка». В ней для девяти участников составляют список боёв по три человека, такой что каждый участник оказывается в одном бою с каждым другим участником ровно один раз.

Участники занумерованы числами от 1 до 9. Вам даётся несколько боёв (троек чисел от 1 до 9), требуется построить минимальную по числу боёв супердевятку, в которой есть все эти бои, или определить, что такой супердевятки не бывает.

Формат входных данных

Первая строка содержит одно целое число n — число заданных боёв ($0 \leq n \leq 84$).

Каждая из последующих n строк содержит по три различных целых числа от 1 до 9 — номера участников соответствующего боя. Гарантируется, что для любых двух боёв есть участник, который участвует в одном из этих боёв и не участвует в другом.

Формат выходных данных

Если решения не существует, выведите -1 . Иначе в первой строке выведите число k — наименьшее число боёв, которое необходимо добавить, а в следующих k строках по три целых числа — номера участников в i -м из дополняющих боёв. Если решений несколько, выведите любое.

Примеры

стандартный ввод	стандартный вывод
3 1 2 3 1 3 4 6 7 8	-1
12 3 2 8 3 9 4 6 7 2 6 8 9 7 1 9 8 1 5 4 7 8 1 6 3 2 1 4 6 4 5 3 5 7 5 9 2	0
0	12 3 2 8 3 9 4 6 7 2 6 8 9 7 1 9 8 1 5 4 7 8 1 6 3 2 1 4 6 4 5 3 5 7 5 9 2

Задача I. Шахматный конь

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Это интерактивная задача.

Шахматная доска 8×8 занумерована стандартным способом: горизонтали — цифрами от 1 до 8, вертикали — строчными латинскими буквами от 'a' до 'h' так, что левый нижний угол — a1, а правый верхний — h8.

На шахматной доске находится конь, изначальная позиция которого неизвестна. Он может ходить на две клетки в одном направлении и на одну клетку в перпендикулярном направлении, оставаясь в пределах доски. Ход коня задаётся следующим образом: сначала указывается смещение по горизонтали x (положительные значения — вправо), затем — смещение по вертикали y (положительные значения — вверх).

Если ход сделать удалось, конь перемещается на новое поле и программа жюри выдаёт строку, состоящую из одного символа '+'. Если же поле назначения находится за пределами доски, конь остаётся на месте и программа жюри выдаёт строку, состоящую из одного символа '-'.

Ваша задача — определить изначальную позицию коня, сделав не более попыток 10 хода (удачных или неудачных).

Протокол взаимодействия

Взаимодействие начинается программа участника выводом запроса. Каждый запрос имеет вид «? $x y$ », где x и y — целые числа и $|xy| = 2$. В ответ на запрос программа жюри выдаёт строку, состоящую из символа '+', если ход сделать удалось, и '-', если ход сделать не удалось.

Когда программа участника готова вывести ответ, она вместо запроса должна вывести «! pos», где pos — изначальная позиция коня, заданная в стандартной шахматной нотации. После этого программа должна завершить своё исполнение.

Пример

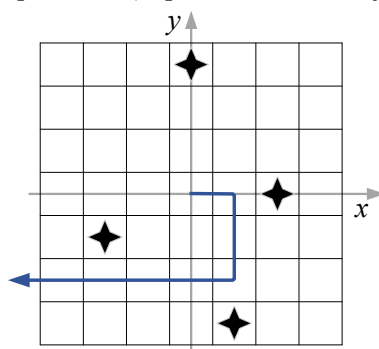
стандартный ввод	стандартный вывод
	? 2 1
+	? -2 -1
+	? 2 -1
-	? -1 2
-	? 1 2
+	! a1

Задача J. Столетний дятел

Ограничение по времени: 2 секунды
 Ограничение по памяти: 512 мегабайт

Карта далёкой-далёкой галактики представляет собой бесконечную плоскость, разбитую на единичные квадраты. Некоторые квадраты заняты звёздами и пролетать через них опасно. Остальные квадраты безопасны.

Космический корабль «Столетний дятел» выходит из червоточины в квадрате $(0, 0)$ и изначально движется вправо (то есть в направлении возрастания первой координаты). После тяжёлого сражения у корабля повреждён двигатель, так что корабль может поворачивать только направо на прямой угол. Корабль управляется автопилотом, который в случае, если следующий по текущему курсу квадрат безопасен, перемещает корабль в него, не тратя энергию. В противном случае автопилот остаётся в текущем квадрате и поворачивает, тратя на это одну единицу энергии.



Требуется определить, сколько единиц энергии потратит корабль, пока одна из его координат не превысит по модулю 10^{10} , или определить, что этого никогда не произойдёт.

Формат входных данных

Первая строка входных данных содержит целое число n — число звёзд в галактике ($0 \leq n \leq 1000$).

Каждая из последующих n строк содержит по два целых числа x_i и y_i — координаты очередной звезды ($-10^9 \leq x_i, y_i \leq 10^9$). Гарантируется, что никакие две звезды не находятся в одном квадрате и что в квадрате $(0, 0)$ звезды нет.

Формат выходных данных

Выведите одно число — количество единиц энергии, которое корабль потратит за время путешествия, если оно закончится, или «oo», если этого никогда не произойдёт.

Примеры

стандартный ввод	стандартный вывод
4 2 0 -2 -1 0 3 1 -3	2
8 1 -1 1 1 1 0 -1 -1 -1 0 -1 1 0 1 0 -1	oo

Задача К. Упавший сервер

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Во Всеберляндской олимпиаде по программированию принимают участие n человек, пронумерованных последовательными целыми числами от 1 до n .

По итогам соревнований участники показали различные результаты и расположились в некотором порядке. Все участники заняли различные места от 1 до n , место i занял участник с номером a_i . Только вот из-за непредвиденного сбоя результаты соревнований были безвозвратно утрачены.

Однако выяснилось, что пока таблица ещё была доступна, два члена жюри независимо сделали пометки относительно результатов тура.

Первый член жюри сделал c_1 записей вида «среди участников, занявших места с l_i по r_i включительно, наименьший номер участника равен m_i ».

Второй член жюри сделал c_2 записей вида «среди участников, занявших места с L_i по R_i включительно, наибольший номер участника равен M_i ».

По имеющимся записям требуется восстановить положение участников a_1, a_2, \dots, a_n или же сообщить, что в записях допущена ошибка.

Если вариантов положения участников несколько, жюри хочет восстановить такое, в котором последовательность a_1, a_2, \dots, a_n лексикографически минимальна.

Формат входных данных

Первая строка входных данных содержит три целых числа n , c_1 и c_2 — количество участников, количество записей, сделанных первым членом жюри, и количество записей, сделанных вторым членом жюри, соответственно ($1 \leq n \leq 50$; $0 \leq c_1 + c_2 \leq 50$).

Последующие c_1 строк задают записи, сделанные первым членом жюри. Каждая такая запись содержит по три целых числа l_i , r_i и m_i ($1 \leq l_i \leq r_i \leq n$; $1 \leq m_i \leq n$).

Последующие c_2 строк задают записи, сделанные вторым членом жюри, в аналогичном формате.

Формат выходных данных

Если данные противоречивы, выведите -1 . Иначе выведите n целых чисел a_1, a_2, \dots, a_n — положение участников, удовлетворяющее всем записям и пожеланию жюри, где a_i — это номер участника, занявшего i -е место.

Примеры

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

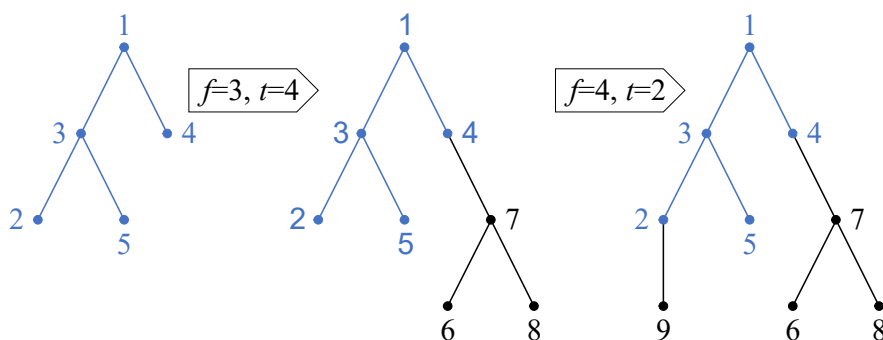
Задача L. Развитие города

Ограничение по времени: 3 секунды
Ограничение по памяти: 512 мегабайт

Город Байтсбург состоит из исторической части и новой части. Историческая часть Байтсбурга представляет собой дерево из n площадей, соединённых $n - 1$ проспектами. Площади занумерованы последовательными целыми числами от 1 до n . Главная площадь города — вершина 1 — является корнем дерева.

Изначально город состоял только из исторической части. Каждый год город развивается следующим образом. Пусть в начале года в городе m площадей.

- Выбирается какая-то площадь f в исторической части и какая-то площадь t среди уже построенных на данный момент (в исторической или в новой части).
- Поддерево T исторической части с корнем в вершине f целиком копируется в новую часть, после чего корень скопированного поддерева соединяется проспектом с площадью t . Все построенные объекты (площади и проспекты) относятся к новой части — историческая часть города остаётся неизменной.
- Пусть поддерево T состоит из k площадей. Тогда новые площади получают номера от $m + 1$ до $m + k$, при этом если в поддереве T номер площади i меньше номера площади j , то номер площади i' , соответствующей площади i , меньше номера площади j' , соответствующей площади j .



Вам дана конфигурация исторической части города и данные о его развитии в течение y лет. Ваша задача — отвечать на запросы вида «найти кратчайшее расстояние между двумя площадями».

Формат входных данных

Первая строка входных данных содержит три целых числа n , y и q — число площадей в исторической части, число лет, в течение которых застраивалась новая часть, и число запросов соответственно ($1 \leq n, y, q \leq 10^5$).

Каждая из последующих $n - 1$ строк содержит по два целых числа a и b — номера двух площадей в исторической части, соединённых очередным проспектом ($1 \leq a, b \leq n$; $a \neq b$). Гарантируется, что конфигурация проспектов и площадей является деревом. Главная площадь, которая является корнем дерева, имеет номер 1.

Каждая из последующих y строк содержит по два целых числа f и t — номер исходной площади в исторической части и номер площади, к которой присоединяется копия ($1 \leq f \leq n$; $t \geq 1$, t не превосходит числа площадей на начало соответствующего года).

Каждая из последующих q строк содержит по два целых числа i и j — номера площадей, расстояние между которыми требуется найти. Пусть M — общее число площадей по прошествии y лет застройки новой части, тогда $1 \leq i, j \leq M$.

Формат выходных данных

Для каждого запроса выведите одно целое число — кратчайшее расстояние между соответствующими площадями.

Пример

стандартный ввод	стандартный вывод
5 2 4	3
1 3	3
1 4	4
3 2	1
3 5	
3 4	
4 2	
5 9	
1 8	
6 3	
4 7	

Задача М. Индекс примечательности

Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Назовём *индексом примечательности* цифровой строки S для заданного простого числа P число различных пар позиций i, j ($1 \leq i \leq j \leq |S|$), для которых число, образованное цифрами, идущими в строке S подряд с i -й по j -ю позицию включительно, делится на P . Число с ведущими нулями считается равным соответствующему числу без ведущих нулей.

Например, для строки 070070 и $P = 13$ соответствующие пары — (1, 1), (1, 5), (1, 6), (2, 5), (2, 6), (3, 3), (3, 4), (4, 4) и (6, 6). Таким образом, её индекс примечательности равен 9.

Задана цифровая строка T и простое число P . Требуется ответить на q запросов вида «найти индекс примечательности для подстроки T с позиции l по позицию r включительно».

Формат входных данных

Первая строка содержит одно простое число P ($2 \leq P \leq 10^9 + 7$). Вторая строка содержит цифровую строку T ($1 \leq |T| \leq 10^5$). Третья строка содержит одно целое число q — число запросов ($1 \leq q \leq 10^5$).

Каждая из последующих q строк задаёт один запрос и содержит два целых числа l и r — левую и правую границу подстроки, индекс примечательности которой интересуется ($1 \leq l \leq r \leq |T|$).

Формат выходных данных

Для каждого запроса выведите на отдельной строке одно целое число — индекс примечательности соответствующей подстроки.

Пример

стандартный ввод	стандартный вывод
13	9
070070	4
3	0
1 6	
2 5	
2 2	

Задача N. N-интересные числа

Ограничение по времени: 5 секунд
Ограничение по памяти: 512 мегабайт

Целое число $X \geq 2$ считается N -интересным, если в его разложении на простые множители $X = p_1 p_2 \cdots p_k$ наибольший из множителей p_i удовлетворяет условиям $p_i^k \leq N$ и $p_i \leq 127$.

Даны два целых числа N и n . Гарантируется, что N -интересных чисел существует не менее n , и при этом их конечное число.

Найдите n -е по убыванию N -интересное число.

Формат входных данных

Первая строка входных данных содержит два целых числа N и n ($2 \leq N \leq 10^{18}$; $1 \leq n \leq 8 \cdot 10^5$).

Формат выходных данных

Выведите одно целое число — n -е по убыванию N -интересное число.

Пример

стандартный ввод	стандартный вывод
3110 21	1573

Задача О. Пожиратель кактусов

Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Как известно, *кактус* — это неориентированный связный граф без петель и кратных рёбер, каждое ребро которого принадлежит не более, чем одному простому циклу.

Недавно был обнаружен микроорганизм *sactofagi* — пожиратель кактусов.

При попадании на компоненту связности он действует следующим образом:

1. Выделяет t джоулей энергии, где t — количество вершин в компоненте.
2. Выбирает случайную вершину на компоненте и поедает её и все инцидентные ей рёбра.
3. Переносится на все образовавшиеся в итоге компоненты связности.

Требуется вычислить математическое ожидание количества энергии, которая выделится при попадании микроорганизма на заданный кактус и последующем полном уничтожении кактуса.

Формат входных данных

Первая строка входных данных содержит два целых числа n и m — число вершин и рёбер кактуса ($1 \leq n \leq 400$; $1 \leq m \leq n(n-1)/2$).

Каждая из последующих m строк содержит по два целых числа a и b — вершины, соединённые очередным ребром ($1 \leq a, b \leq n$). Гарантируется, что заданный граф является кактусом.

Формат выходных данных

Можно показать, что ответ является рациональным числом. Представьте его в виде несократимой дроби p/q и выведите значение $pq^{-1} \bmod 998\,244\,353$.

Примеры

стандартный ввод	стандартный вывод
4 3 1 4 2 4 3 4	9
5 5 1 2 2 3 3 4 4 5 5 2	798595496

Замечание

Во втором примере ответ равен 13.6, или же $68/5$ в виде несократимой дроби.