

Задача А. Сумма на отрезке

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дан массив из N элементов, нужно научиться находить сумму чисел на отрезке.

Формат входных данных

Первая строка входного файла содержит два целых числа N и K — количество чисел в массиве и количество запросов ($1 \leq N \leq 100\,000$, $0 \leq K \leq 100\,000$). Следующие K строк содержат следующие запросы:

1. $A\ i\ x$ — присвоить i -му элементу массива значение x ($1 \leq i \leq n$, $0 \leq x \leq 10^9$);
2. $Q\ l\ r$ — найти сумму чисел в массиве на позициях от l до r ($1 \leq l \leq r \leq n$).

Изначально в массиве живут нули.

Формат выходных данных

На каждый запрос вида $Q\ l\ r$ нужно вывести единственное число — сумму на отрезке.

Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 5 9 | 0 |
| A 2 2 | 2 |
| A 3 1 | 1 |
| A 4 2 | 2 |
| Q 1 1 | 0 |
| Q 2 2 | 5 |
| Q 3 3 | |
| Q 4 4 | |
| Q 5 5 | |
| Q 1 5 | |

Задача В. Звезды

| | |
|-------------------------|-------------------|
| Имя входного файла: | стандартный ввод |
| Имя выходного файла: | стандартный вывод |
| Ограничение по времени: | 2 секунды |
| Ограничение по памяти: | 256 мегабайт |

Вася любит наблюдать за звездами. Но следить за всем небом сразу ему тяжело. Поэтому он наблюдает только за частью пространства, ограниченной кубом размером $n \times n \times n$. Этот куб поделен на маленькие кубики размером $1 \times 1 \times 1$. Во время его наблюдений могут происходить следующие события:

1. В каком-то кубике появляются или исчезают несколько звезд.
2. К нему может заглянуть его друг Петя и поинтересоваться, сколько видно звезд в части пространства, состоящей из нескольких кубиков.

Формат входных данных

Первая строка входного файла содержит натуральное число $1 \leq n \leq 128$. Координаты кубиков — целые числа от 0 до $n - 1$. Далее следуют записи о происходивших событиях по одной в строке. В начале строки записано число m . Если m равно:

- 1, то за ним следуют 4 числа — x, y, z ($0 \leq x, y, z < N$) и k ($-20000 \leq k \leq 20000$) — координаты кубика и величина, на которую в нем изменилось количество видимых звезд;
- 2, то за ним следуют 6 чисел — $x_1, y_1, z_1, x_2, y_2, z_2$ ($0 \leq x_1 \leq x_2 < N, 0 \leq y_1 \leq y_2 < N, 0 \leq z_1 \leq z_2 < N$), которые означают, что Петя попросил подсчитать количество звезд в кубиках (x, y, z) из области: $x_1 \leq x \leq x_2, y_1 \leq y \leq y_2, z_1 \leq z \leq z_2$;
- 3, то это означает, что Васе надоело наблюдать за звездами и отвечать на вопросы Пети. Эта запись встречается во входном файле только один раз и будет последней.

Количество записей во входном файле не больше 100 002.

Формат выходных данных

Для каждого Петиного вопроса выведите искомое количество звезд.

Пример

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

Задача С. Сокровища

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

Дочь короля Флатландии собирается выйти за прекрасного принца. Принц хочет подарить принцессе сокровища, но он не уверен какие именно бриллианты из своей коллекции выбрать.

В коллекции принца n бриллиантов, каждый характеризуется весом w_i и стоимостью v_i . Принц хочет подарить наиболее дорогие бриллианты, однако король умен и не примет бриллиантов суммарного веса больше R . С другой стороны, принц будет считать себя жадным всю оставшуюся жизнь, если подарит бриллиантов суммарным весом меньше L .

Помогите принцу выбрать набор бриллиантов наибольшей суммарной стоимости, чтобы суммарный вес был в отрезке $[L, R]$.

Формат входных данных

Первая строка содержит число n ($1 \leq n \leq 32$), L и R ($0 \leq L \leq R \leq 10^{18}$). Следующие n строк описывают бриллианты и содержит по два числа — вес и стоимость соответствующего бриллианта ($1 \leq w_i, v_i \leq 10^{15}$).

Формат выходных данных

Первая строка вывода должна содержать k — количество бриллиантов, которые нужно подарить принцессе. Вторая строка должна содержать номера даримых бриллиантов.

Бриллианты нумеруются от 1 до n в порядке появления во входных данных.

Если составить подарок принцессе невозможно, то выведите 0 в первой строке вывода.

Пример

| <code>dowry.in</code> | <code>dowry.out</code> |
|-----------------------|------------------------|
| 3 6 8 | 1 |
| 3 10 | 2 |
| 7 3 | |
| 8 2 | |

Задача D. Небоскрёбы

| | |
|-------------------------|-------------------|
| Имя входного файла: | стандартный ввод |
| Имя выходного файла: | стандартный вывод |
| Ограничение по времени: | 1 секунда |
| Ограничение по памяти: | 64 мегабайта |

Вы когда-нибудь мечтали стать главным героем компьютерной игры? Главный герой этой истории, Бранимир, мечтает сейчас именно об этом.

Мир в мечте Бранимира состоит из N небоскребов, пронумерованных слева направо. Для i -го небоскреба, известна его высота H_i и количество золотых монет G_i на крыше этого небоскреба. Игра начинается с прыжка на любой из небоскребов и состоит из нескольких ходов. На каждом ходу Бранимир может прыгнуть на любой небоскрёб, находящийся **справа** от него, но так, чтобы высота нового небоскрёба была **не меньше** того небоскрёба, на котором сейчас сидит Бранимир. Оказавшись на крыше небоскреба, Бранимир собирает все золотые монеты на ней. Бранимир может закончить игру после любого количества шагов (возможно, нулевого), но он должен собрать не менее K золотых монет, чтобы перейти на следующий уровень.

Бранимир хочет узнать, сколько существует способов сыграть в эту игру так, чтобы перейти на следующий уровень. Две игры называются разными, если существует небоскрёб который был посещен в одной игре, но не был посещён в другой.

Формат входных данных

Первая строка содержит 2 натуральных числа N и K ($1 \leq N \leq 40$, $1 \leq K \leq 4 \cdot 10^{10}$) — число небоскребов и количество монет, которые надо набрать соответственно.

Следующие N строк содержат информацию о небоскрёбах. В i -й строке даны 2 числа H_i и G_i ($1 \leq H_i, G_i \leq 10^9$) — высота и количество монет на i -м небоскрёбе.

Формат выходных данных

В единственной строке вывода выведите число возможных игр, в которых Бранимир сможет пройти на следующий уровень.

Система оценки

Решение, корректно работающее при $n \leq 20$ будет оцениваться в 40 баллов.

Примеры

| стандартный ввод | стандартный вывод |
|------------------------------------|-------------------|
| 4 6 2 1 6 3 7 2 5 6 | 3 |
| 2 7 4 6 3 5 | 0 |
| 4 15 5 5 5 12 6 10 2 1 | 4 |

Задача Е. Потерянный массив

| | |
|-------------------------|-------------------|
| Имя входного файла: | стандартный ввод |
| Имя выходного файла: | стандартный вывод |
| Ограничение по времени: | 1 секунда |
| Ограничение по памяти: | 256 мегабайт |

Маленькому Леше улыбнулась удача: ему удалось раздобыть очень редкий и необычный массив A , состоящий из n целых чисел. Заполучив в свои руки такой ценный предмет, мальчик сразу же отнес его в свою подземную лабораторию, намереваясь постичь тайны устройства массива.

Как опытный исследователь, Леша решил не спешить с проведением жутких экспериментов над бедным массивом, а сперва понаблюдать за ним. Он записал на листик бумаги результаты m наблюдений за массивом. Каждое наблюдение описывается тремя числами l_i , r_i и s_i , которые означают, что сумма элементов по модулю 2 на подотрезке $[l_i, r_i]$ массива равна s_i . Леша подготовил все необходимое для проведения $m+1$ -го наблюдения, но тут он услышал, что мама зовет его обедать.

По возвращении в свою лабораторию, мальчик увидел страшное: массив исчез. Совсем. Все надежды мальчика на получение Нобелевской премии в области исследования массивов рассыпались на глазах. Но потом Леша вспомнил, что у него все же осталась некоторая информация о массиве. Основываясь на записанных фактах, он решил построить компьютерную модель массива и проводить эксперименты над ней. Конечно, мальчик понимал, что он не сможет однозначно восстановить потерянный массив, потому что существует бесконечно много массивов, удовлетворяющих записанным фактам.

Все мы знаем, что операции с большими числами занимают много времени, а если еще к тому же числа становятся отрицательными... Чтобы сделать свою компьютерную модель как можно более быстрой, Леша решил построить массив, удовлетворяющий известным фактам, элементы которого неотрицательны, а их сумма минимально возможная.

Помогите ему сделать это!

Формат входных данных

Первая строка входных данных содержит числа n и m - количество элементов в массиве и количество фактов. Следующие m строк содержат числа l_i , r_i и s_i - описание i -го факта.

$$\begin{aligned} 1 &\leq n \leq 40 \\ 0 &\leq m \leq 100 \\ 1 &\leq l_i \leq r_i \leq n \\ 0 &\leq s_i \leq 1 \\ 0 &\leq a_i \leq 10^9 \end{aligned}$$

Формат выходных данных

Вывод должен содержать n целых чисел a_i — элементы восстановленного массива. Если существует несколько возможных ответов, выведете лексикографически минимальных среди них. Массив X лексикографически меньше массива Y если существует такой индекс $k \leq |X|$, что $x_i = y_i$ для всех $i < k$ и $x_k < y_k$.

Пример

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

Задача F. Максимальная подпоследовательность

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вам дан массив a , состоящий из n целых чисел и целое число m . Выберите последовательность позиций b_1, b_2, \dots, b_k ($1 \leq b_1 < b_2 < \dots < b_k \leq n$) такую, чтобы значение $\sum_{i=1}^k a_{b_i} \bmod m$ было максимально. Выбранная подпоследовательность может быть пустой.

Подсчитайте максимальное возможное значение $\sum_{i=1}^k a_{b_i} \bmod m$.

Формат входных данных

В первой строке записаны два целых числа n и m ($1 \leq n \leq 35$, $1 \leq m \leq 10^9$).

Во второй строке записаны n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$).

Формат выходных данных

Выведите максимальное возможное значение $\sum_{i=1}^k a_{b_i} \bmod m$.

Примеры

| стандартный ввод | стандартный вывод |
|--------------------|-------------------|
| 4 4 5 2 4 1 | 3 |
| 3 20 199 41 299 | 19 |

Замечание

В первом примере можно выбрать последовательность $b = 1, 2$, чтобы сумма $\sum_{i=1}^k a_{b_i}$ была равна 7 (равна 3 по модулю 4).

Во втором примере можете выбрать последовательность $b = 3$.

Задача G. Необычная сортировка

| | |
|-------------------------|-------------------|
| Имя входного файла: | стандартный ввод |
| Имя выходного файла: | стандартный вывод |
| Ограничение по времени: | 3 секунды |
| Ограничение по памяти: | 256 мегабайт |

Вам дана последовательность различных целых чисел a_1, a_2, \dots, a_n . Все числа в последовательности лежат в отрезке $[0, 2^k - 1]$, где k дано.

Давайте определим функцию $f(x)$ для числа x , которое лежит в отрезке $[0, 2^k - 1]$ как количество инверсий в последовательности $a_1 \oplus x, a_2 \oplus x, \dots, a_n \oplus x$.

Теперь давайте отсортируем все целые числа из отрезка $[0, 2^k - 1]$ по возрастанию значений функции f , а затем по возрастанию самих чисел.

Вам дана позиция p . Найдите p -е число в этом порядке сортировки всех целых чисел из отрезка $[0, 2^k - 1]$.

Формат входных данных

В первой строке находится единственное целое число t ($1 \leq t \leq 20$) — количество тестовых случаев. Далее находится описание t тестовых случаев в следующем формате:

Первая строка содержит три целых числа n, k, p , разделенных пробелом ($1 \leq n \leq 10^6, 1 \leq k \leq 30, 1 \leq p \leq 2^k$) — количество чисел в последовательности, параметр k и заданная позиция.

В следующей строке находятся n различных целых чисел a_1, a_2, \dots, a_n , разделенных пробелами ($0 \leq a_i < 2^k$).

Сумма всех n во всех тестовых случаях не превосходит 10^6 .

Формат выходных данных

Для каждого тестового случая выведите единственное целое число — p -е целое число в порядке сортировки всех целых чисел из отрезка $[0, 2^k - 1]$, описанном в условии задачи.

Пример

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

Задача Н. Мать драконов

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.3 секунд
Ограничение по памяти: 256 мегабайт

В Королевстве Ланнистеров n замков и несколько стен, соединяющих два замка, никакие два замка не соединены более, чем одной стеной, ни одна стена не соединяет замок с собой.

Сир Джейме Ланнистер узнал, что Дейенерис Таргариен собирается атаковать его королевство. Он хочет защитить свои владения. У него есть k литров странной жидкости. Он хочет распределить эту жидкость между замками так, чтобы каждый замок содержал некоторое количество жидкости (возможно, нулевое или нецелое количество литров). После этого стабильность стены, соединяющей замки a и b , содержащие x и y литров жидкости, соответственно, равна $x \cdot y$.

Ваша задача — найти максимальную возможную сумму стабильностей стен, которую Сир Джейме Ланнистер сможет достичь

Формат входных данных

Первая строка содержит два целых числа n и k ($1 \leq n \leq 40$, $1 \leq k \leq 1000$).

Далее следует n строк. В i -й из них содержится n целых чисел $a_{i,1}, a_{i,2}, \dots, a_{i,n}$ ($a_{i,j} \in \{0, 1\}$). Если замки i и j соединены стеной, $a_{i,j} = 1$. В противном случае оно равно 0.

Гарантируется, что $a_{i,j} = a_{j,i}$ и $a_{i,i} = 0$ для всех $1 \leq i, j \leq n$.

Формат выходных данных

Выведите одно число — максимальную возможную сумму стабильностей стен, которую Сир Джейме Ланнистер сможет достичь.

Ваш ответ будет считаться правильным, если его абсолютная или относительная точность не превосходит 10^{-6} .

А именно, если ваш ответ равен a , а ответ жюри равен b , то ваш ответ будет зачтен, если $\frac{|a-b|}{\max(1,b)} \leq 10^{-6}$.

Примеры

| стандартный ввод | стандартный вывод |
|---|-------------------|
| 3 1 0 1 0 1 0 0 0 0 0 | 0.250000000000 |
| 4 4 0 1 0 1 1 0 1 0 0 1 0 1 1 0 1 0 | 4.000000000000 |

Замечание

В первом примере, если замки 1, 2, 3 содержат 0.5, 0.5, 0 литров жидкости, соответственно, ответ равен 0.25.

Во втором примере, если замки 1, 2, 3, 4 содержат 1.0, 1.0, 1.0, 1.0 литров жидкости, ответ равен 4.0.

Задача I. Катый ноль

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Реализуйте эффективную структуру данных, позволяющую изменять элементы массива и вычислять индекс k -го слева нуля на данном отрезке в массиве.

Формат входных данных

В первой строке вводится одно натуральное число N ($1 \leq N \leq 200\,000$) — количество чисел в массиве. Во второй строке вводятся N чисел от 0 до 100 000 — элементы массива. В третьей строке вводится одно натуральное число M ($1 \leq M \leq 200\,000$) — количество запросов. Каждая из следующих M строк представляет собой описание запроса. Сначала вводится одна буква, кодирующая вид запроса (s — вычислить индекс k -го нуля, u — обновить значение элемента). Следом за s вводится три числа — левый и правый концы отрезка и число k ($1 \leq k \leq N$). Следом за u вводятся два числа — номер элемента и его новое значение.

Формат выходных данных

Для каждого запроса s выведите результат. Все числа выводите в одну строку через пробел. Если нужного числа нулей на запрашиваемом отрезке нет, выводите -1 для данного запроса.

Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 5 | 4 |
| 0 0 3 0 2 | |
| 3 | |
| u 1 5 | |
| u 1 0 | |
| s 1 5 3 | |

Замечание

TL для Python 8 секунд

Задача J. Разреженные таблицы

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дан массив из n чисел. Требуется написать программу, которая будет отвечать на запросы следующего вида: найти минимум на отрезке между u и v включительно.

Формат входных данных

В первой строке заданы три натуральных числа n , m ($1 \leq n \leq 10^5$, $1 \leq m \leq 10^7$) и a_1 ($0 \leq a_1 < 16\,714\,589$) — количество элементов в массиве, количество запросов и первый элемент массива соответственно. Вторая строка содержит два натуральных числа u_1 и v_1 ($1 \leq u_1, v_1 \leq n$) — первый запрос.

Для того, чтобы размер ввода был небольшой, массив и запросы генерируются.

Элементы a_2, a_3, \dots, a_n задаются следующей формулой:

$$a_{i+1} = (23 \cdot a_i + 21563) \bmod 16714589.$$

Например, при $n = 10$, $a_1 = 12345$ получается следующий массив: $a = (12345, 305498, 7048017, 11694653, 1565158, 2591019, 9471233, 570265, 13137658, 1325095)$.

Запросы генерируются следующим образом:

$$u_{i+1} = ((17 \cdot u_i + 751 + r_i + 2i) \bmod n) + 1,$$
$$v_{i+1} = ((13 \cdot v_i + 593 + r_i + 5i) \bmod n) + 1,$$

где r_i — ответ на запрос номер i .

Обратите внимание, что u_i может быть больше, чем v_i .

Формат выходных данных

В выходной файл выведите u_m , v_m и r_m (последний запрос и ответ на него).

Примеры

| стандартный ввод | стандартный вывод |
|-------------------|-------------------|
| 10 8 12345 3 9 | 5 3 1565158 |

Замечание

Можно заметить, что массивы u , v и r можно не сохранять в памяти полностью.

Запросы и ответы на них выглядят следующим образом:

| i | u_i | v_i | r_i |
|-----|-------|-------|---------|
| 1 | 3 | 9 | 570265 |
| 2 | 10 | 1 | 12345 |
| 3 | 1 | 2 | 12345 |
| 4 | 10 | 10 | 1325095 |
| 5 | 5 | 9 | 570265 |
| 6 | 2 | 1 | 12345 |
| 7 | 3 | 2 | 305498 |
| 8 | 5 | 3 | 1565158 |

Эта задача скорее всего не решается стандартными интерпретаторами Python 2 и Python 3. Используйте соответствующие компиляторы PyPy.

Задача К. LCA - 2

Имя входного файла: lca2.in
Имя выходного файла: lca2.out
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

256 мегабайт

Задано подвешенное дерево, содержащее n вершин, пронумерованных от 0 до $n - 1$. Требуется ответить на m запросов о наименьшем общем предке для пары вершин.

Запросы генерируются следующим образом. Заданы числа a_1, a_2 и числа x, y и z .

Числа a_3, \dots, a_{2m} генерируются следующим образом: $a_i = (x \cdot a_{i-2} + y \cdot a_{i-1} + z) \bmod n$. Первый запрос имеет вид $\langle a_1, a_2 \rangle$. Если ответ на $i - 1$ -й запрос равен v , то i -й запрос имеет вид $\langle (a_{2i-1} + v) \bmod n, a_{2i} \rangle$.

Формат входных данных

Первая строка содержит два числа: n ($1 \leq n \leq 100\,000$) и m ($1 \leq m \leq 10\,000\,000$). Корень дерева имеет номер 0. Вторая строка содержит $n - 1$ целых чисел, i -е из этих чисел это предок вершины i

Третья строка содержит целые числа a_1 и a_2 ($0 \leq a_i \leq n - 1$).

Четвёртая строка содержит три целых числа: x, y и z ($0 \leq x, y, z \leq 10^9$)

Формат выходных данных

Выведите в выходной файл сумму номеров вершин — ответов на все запросы.

Примеры

| lca2.in | lca2.out |
|----------------------------|----------|
| 3 2 0 1 2 1 1 1 0 | 2 |
| 1 2 0 0 1 1 1 | 0 |

Задача L. Другие разреженные таблицы

| | |
|-------------------------|-------------------|
| Имя входного файла: | стандартный ввод |
| Имя выходного файла: | стандартный вывод |
| Ограничение по времени: | 1 секунда |
| Ограничение по памяти: | 256 мегабайт |

Дан массив из n чисел и модуль m (необязательно простой). Требуется написать программу, которая будет отвечать на запросы следующего вида: найти произведение чисел на отрезке между u и v включительно по модулю m .

Формат входных данных

В первой строке входного файла даны четыре натуральных числа n , m , q ($1 \leq n \leq 10^5$, $1 \leq m \leq 10^9$, $1 \leq q \leq 10^7$) и a_1 ($0 \leq a_1 < 16714589$) — количество элементов в массиве, модуль, количество запросов и первый элемент массива соответственно. Вторая строка содержит два натуральных числа u_1 и v_1 ($1 \leq u_1, v_1 \leq n$) — первый запрос.

Элементы a_2, a_3, \dots, a_n задаются следующей формулой:

$$a_{i+1} = (23 \cdot a_i + 21563) \bmod 16714589.$$

Например, при $n = 10$, $a_1 = 12345$ получается следующий массив: $a = (12345, 305498, 7048017, 11694653, 1565158, 2591019, 9471233, 570265, 13137658, 1325095)$.

Запросы генерируются следующим образом:

$$\begin{aligned} u_{i+1} &= ((17 \cdot u_i + 751 + ans_i + 2i) \bmod n) + 1, \\ v_{i+1} &= ((13 \cdot v_i + 593 + ans_i + 5i) \bmod n) + 1, \end{aligned}$$

где ans_i — ответ на запрос номер i .

Обратите внимание, что u_i может быть больше, чем v_i .

Формат выходных данных

В выходной файл выведите u_q , v_q и ans_q (последний запрос и ответ на него).

Пример

| стандартный ввод | стандартный вывод |
|------------------------|-------------------|
| 10 5677 8 12345 3 9 | 5 3 1631 |

Задача М. Прямоугольники

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Когда-то тут была легенда про вёдра, но её съели.

Жюри олимпиады

Есть таблица T размера $N \times M$. Элементами таблицы являются прямоугольники T_{ij} , где $0 \leq i < N$ и $0 \leq j < M$. Прямоугольник T_{ij} задаётся четвёркой чисел $(x_1^{ij}, y_1^{ij}, x_2^{ij}, y_2^{ij})$, где (x_1^{ij}, y_1^{ij}) и (x_2^{ij}, y_2^{ij}) — координаты противоположных углов прямоугольника. Стороны прямоугольника параллельны осям координат.

Далее вам поступают запросы. Каждый запрос состоит из четырёх чисел: (r_1, c_1, r_2, c_2) . Ответом на такой запрос является площадь фигуры, являющейся пересечением всех прямоугольников T_{ij} таких, что $\min(r_1, r_2) \leq i \leq \max(r_1, r_2)$ и $\min(c_1, c_2) \leq j \leq \max(c_1, c_2)$. Запросов очень много, поэтому мы просим вас вывести сумму ответов на все запросы по модулю $10^9 + 7$.

Формат входных данных

В первой строке записаны два целых числа N и M — размеры таблицы T ($1 \leq N, M \leq 127$). Далее в N строках описывается таблица T : в $(i + 1)$ -й строке $(j + 1)$ -я четвёрка чисел $x_1^{ij} y_1^{ij} x_2^{ij} y_2^{ij}$ описывает прямоугольник T_{ij} . Гарантируется, что $|x_k^{ij}|, |y_k^{ij}| \leq 10^6$.

Дальше в отдельной строке записано четыре числа. Первое из них, число Q — количество запросов ($1 \leq Q \leq 5 \cdot 10^6$). Следующие три числа — это A, B, v_0 ($0 \leq A, B, v_0 < 10^9 + 7$). При помощи этих чисел генерируется бесконечная последовательность $\{v_i\}$ по правилу $v_i = (A \cdot v_{i-1} + B) \bmod (10^9 + 7)$.

После этого k -й запрос (запросы нумеруются с единицы) задаётся следующей четвёркой чисел: $(v_{4k-3} \bmod N, v_{4k-2} \bmod M, v_{4k-1} \bmod N, v_{4k} \bmod M)$.

Формат выходных данных

Выведите сумму ответов на все запросы по модулю $10^9 + 7$.

Примеры

| стандартный ввод | стандартный вывод |
|---|-------------------|
| 2 2 0 0 2 2 1 1 3 3 0 3 2 1 1 2 3 0 1 500000003 4 2 | 1 |
| 3 2 8 -1 -7 6 6 8 9 10 -4 -10 4 9 -3 -8 6 9 -2 -9 3 8 -5 7 7 3 5 303164476 273973578 65779139 | 85 |

Замечание

В первом примере запрос имеет вид $(1, 0, 0, 1)$, то есть это запрос ко всей таблице. Пересечением всех прямоугольников является квадрат с углами в точках $(1, 1)$ и $(2, 2)$. Его площадь равна 1.

Во втором примере запросы имеют вид $(0, 1, 1, 1)$, $(1, 0, 2, 0)$, $(0, 0, 2, 1)$, $(0, 1, 1, 1)$, $(0, 1, 0, 0)$. На второй запрос ответ — 85, на остальные — 0.