

Example

стандартный ввод	стандартный вывод
<pre>game (3,1) { art(0); if (a[1] >= 1) art (1); else if (a[1] >= 2) art(2); if (a[2] >= 1) if (a[3] >= 1) art(3); else art(4); {art(0)}; }</pre>	3

Note

Артефакты 0, 1 и 4 являются достижимыми.

Problem B. Brackets and Dots

Input file: **стандартный ввод**
Output file: **стандартный вывод**
Time limit: 1 секунда
Memory limit: 256 мегабайт

Андрей любит строки, которые состоят из круглых скобок, но еще больше он любит заменять скобки на точки.

У него есть строка длины N , к которой он применяет M операций. Каждая операция задается двумя натуральными числами l_i и r_i , в результате операции в подстроке $s_{l_i}s_{l_i+1}, \dots, s_{r_i}$ самая длинная правильная скобочная подпоследовательность (далее ПСП) заменяется на точки (каждая скобочка заменяется на точку).

Андрей хочет себя проверить и просит вас помочь ему: подсчитать сколько скобок после каждой операции были заменены на точки.

Если в подстроке есть несколько ПСП максимальной длины, то выбирается наименьшая. Правило сравнения следующие: пусть a_1, a_2, \dots, a_l — позиции открывающихся скобок первой ПСП, а c_1, c_2, \dots, c_l — позиции открывающихся скобок второй ПСП. Считается, что первая ПСП меньше второй, если существует k ($1 \leq k \leq l$), такое что $a_i = c_i$ ($1 \leq i < k$) и $a_k > c_k$.

Если позиции открывающихся скобок обеих ПСП совпали, то сравниваются позиции закрывающихся скобок по следующему правилу: пусть b_1, b_2, \dots, b_l — позиции закрывающихся скобок первой ПСП, а d_1, d_2, \dots, d_l — позиции закрывающихся скобок второй ПСП. Считается, что первая ПСП меньше второй, если существует k ($1 \leq k \leq l$), такое что $b_i = d_i$ ($1 \leq i < k$) и $b_k < d_k$.

Input

На первой строке задается исходная строка S длины N ($1 \leq N \leq 5 \times 10^5$), которая состоит из открывающихся и закрывающихся круглых скобок.

На второй строке задается натуральное число M ($1 \leq M \leq 5 \times 10^5$) — количество операций.

В следующих M строках задаются операции, каждая строка состоит из двух натуральных чисел l_i и r_i ($1 \leq l_i \leq r_i \leq N$) — левая и правая граница подстроки соответственно.

Output

Для каждой операции на отдельной строке выведите единственное число — число скобок, которые были заменены на точки.

Example

стандартный ввод	стандартный вывод
((((()())))	4
3	2
4 9	4
2 8	
1 10	

Problem C. Crossword

Input file: *стандартный ввод*
Output file: *стандартный вывод*
Time limit: 2 секунды
Memory limit: 256 мегабайт

Составление кроссвордов — непростая задача. Помимо того, чтобы соблюсти баланс по сложности слов, необходимо уместить слова в ограниченную площадь.

Но и это не всё! Пустого места должно быть не слишком много, но и не слишком мало, чтобы его можно было заполнить рекламой.

Вам даны четыре слова. Сколькими способами можно составить из них кроссворд, чтобы внутри образовалась пустая область в виде прямоугольника, ни одна из сторон которого не равна 0? Каждое слово должно пересекаться с ровно двумя другими словами.

Input

Ввод содержит 4 слова, находящиеся в отдельных строках.

Каждое слово имеет длину от 3 до 100 и состоит только из малых букв латинского алфавита.

Все слова различны.

Output

Выведите искомое количество вариантов.

Examples

стандартный ввод	стандартный вывод
aaa axa aya aza	24
aaaa abba baab bbbb	40

Problem D. Digit

Input file: *стандартный ввод*
Output file: *стандартный вывод*
Time limit: 1 секунда
Memory limit: 256 мебибайт

Проснувшись ранним утром, Сашка понял, что давно не проводывал своего старого друга - Витальку. Решив не откладывать это дело на следующий день, он начал собираться в гости. Сашка уже стоял на пороге своей квартиры, как понял, что у него нет подарка! Зная слабость Витальки к цифрам, он решил подарить ему самую часто встречающуюся цифру в записи чисел от 1 до N , причем если таких несколько, было решено взять максимальную. Сможете ли Вы определить, какую цифру подарил Сашка своему другу?

Input

В единственно строке задано целое число N ($1 \leq N \leq 10^{100000}$).

Output

Выведите цифру, которую подарил Сашка.

Examples

стандартный ввод	стандартный вывод
100	1
99	9

Problem E. Enormous Table

Input file: *стандартный ввод*
Output file: *стандартный вывод*
Time limit: 1 секунда
Memory limit: 256 мегабайт

В бесконечной таблице по номеру строки и столбца определить элемент.

```
1 2 6 7 15 ...
3 5 8 14 ...
4 9 13 ...
10 12 ...
11 ...
```

Input

Два натуральных числа a, b — номер строки и столбца соответственно. Числа не превосходят 10^9 .

Output

Выведите одно натуральное число — ответ на задачу.

Example

стандартный ввод	стандартный вывод
2 3	8

Problem F. Funny Language

Input file: *стандартный ввод*
Output file: *стандартный вывод*
Time limit: 1 секунда
Memory limit: 256 мебибайт

В тау-китянском языке M слов, каждое слово представляет собой последовательность букв — целых чисел в интервале от 1 до $8 \cdot 10^4$. Алиса занумеровала слова последовательными целыми числами от 1 до M , после чего делает следующие действия: составляет предложение, в которое бы входило каждое тау-китянское слово по одному разу (в тау-китянском языке слова в предложении не разделяются) и записывает сумму НОД по всем различным подотрезкам получившейся последовательности (подотрезок задаётся двумя позициями — началом и концом, при этом конец не может быть левее начала).

После того, как будут перебраны все $M!$ перестановок, Алиса складывает все записанные числа. Требуется вычислить остаток от деления итоговой суммы на $10^9 + 7$.

Input

Первая строка входных данных содержит одно целое число M ($1 \leq M \leq 8 \cdot 10^4$).

Далее следуют M строк. Каждая строка начинается целым числом S Then followed M lines. Each line start with a integer S ($1 \leq S \leq 8 \cdot 10^4$), then followed S integers. The sum of S does not exceed $8 \cdot 10^4$. All integers are between 1 and $8 \cdot 10^4$.

Output

Print the answer modulo $10^9 + 7$.

Example

стандартный ввод	стандартный вывод

Problem G. Game of Tic-Tac-Toe

Input file: **стандартный ввод**
Output file: **стандартный вывод**
Time limit: 1 секунда
Memory limit: 256 мегабайт

Это интерактивная задача.

В недалёком будущем OpenAI бот уже выигрывает у человека почти во всех играх. Чтобы не дать боту захватить мир вам нужно написать программу, которая даст отпор при игре в классические крестики-нолики.

Напомним правила этой игры: игроки ходят по очереди, выставляя на клетчатом поле 3 на 3 свой символ (X или O — заглавные латинские буквы). Ходить можно только в свободную клетку. Если после хода образовалось 3 одинаковых символа в ряд (по вертикали, горизонтали или диагонали), игрок, которому принадлежит символ, выигрывает. Если такого не произошло, но при этом нет свободных клеток, то в игре объявляется ничья.

OpenAI бот всегда играет за нолики, Вы — за крестики. Ваша цель — не проиграть боту, glhf!

Interaction Protocol

В самом начале в первой строке ввода задаётся, кто ходит первым.

Далее происходит общение. Если ход за ботом, то он сообщает вам координаты клетки, в которую сделал ход, в виде пары целых чисел r и c , означающих номер строки и столбца. Строки нумеруются от 1 до 3 сверху вниз, а столбцы от 1 до 3 слева направо. Каждый ход выводится в отдельной строке, числа разделены одним пробелом. Гарантируется, что ходы бота корректны.

Если ход за вами, то Вы должны вывести в аналогичном формате координаты клетки хода.

Как только игра заканчивается, вам немедленно сообщается результат игры в виде строки «WIN» (вы выиграли), «LOSE» (вы проиграли) или «DRAW» (ничья).

Если вы проиграли или один из ваших ходов является некорректным, вам будет засчитан неправильный ответ на тест.

Examples

стандартный ввод	стандартный вывод
X	1 1
1 3	2 1
2 3	3 1
WIN	
O	3 3
1 1	1 3
3 1	2 1
2 3	3 2
1 2	
2 2	
DRAW	

Note

Для корректной работы программы после каждой операции вывода данных вам необходимо выводить перевод строки, а также очищать буфер вывода, то есть делать следующие операции:

- В языке Pascal: `flush(output);`
- В C/C++: `fflush(stdout)` или `cout.flush();`

- В Java: **System.out.flush()**;
- В Python: **sys.stdout.flush()** из библиотеки **sys**;
- В C#: **Console.Out.Flush()**;

Problem H. Hill and Subhill

Input file: **стандартный ввод**
Output file: **стандартный вывод**
Time limit: 1 секунда
Memory limit: 256 мегабайт

Определим *гору* как множество точек

$$P(n) = \{(x, y, z) | 1 \leq z \leq y \leq x \leq n\}$$

а *подгору* как множество точек

$$sP(x, y, z, a) = \{(x + i, y + j, z + k) | 0 \leq k \leq j \leq i < a\}$$

Вам дана гора $P(N)$, каждой целой точке горы $P(N)$ изначально присвоено значение 0.

Вы должны выполнить M операций $A(x_i, y_i, z_i, a_i)$ — прибавление единицы к значению каждой целой точки в подгоре $sP(x_i, y_i, z_i, a_i)$.

После чего вы должны ответить на Q запросов $Q(x_j, y_j, z_j, a_j)$, запрашивающих сумму значений всех целых точек в подгоре $sP(x_j, y_j, z_j, a_j)$.

Input

Первая строка ввода содержит три целых числа N , M и Q ($1 \leq N \leq 100$, $1 \leq M \leq 10^5$, $1 \leq Q \leq 10^5$).

Далее следуют M строк, задающих операции прибавления в порядке, в котором они выполняются. Каждая строка содержит четыре целых числа x_i, y_i, z_i, a_i .

Далее следуют Q строк, задающих запросы в том порядке, в котором требуется на них отвечать. Каждая строка содержит четыре целых числа x_j, y_j, z_j, a_j .

Гарантируется, что параметры запросов и операций корректны.

Output

Для каждого запроса в новой строке выведите одно целое число — ответ на запрос.

Example

стандартный ввод	стандартный вывод
2 1 2	4
1 1 1 2	1
1 1 1 2	
2 1 1 1	

Problem I. It is panic?

Input file: *стандартный ввод*
Output file: *стандартный вывод*
Time limit: 1 секунда
Memory limit: 256 мебибайт

Ваш одноклассник Вася устроился в службу экстренного реагирования МЧС. Когда в городе что-то происходит, люди с места происшествия отправляют в эту службу SMS-сообщения. Если сообщение имеет вид «AAA...!!!!», то есть его можно разбить на две непустые половинки так, что левая состоит только из заглавных букв A, а правая — только из восклицательных знаков, то оно считается *паническим*, и на место этого происшествия группа реагирования отправляется в первую очередь. Если сообщение не паническое, то группа реагирования отправляется на место этого происшествия только тогда, когда нет других происшествий с паническими сообщениями.

В октябре как всегда внезапно выпал снег, в связи с этим количество происшествий, а значит, и SMS-сообщений, которые нужно обработать Васе, резко увеличилось. Помогите Васе и напишите программу, которая определяет, является SMS-сообщение паническим или нет.

Input

Ввод содержит непустую строку из заглавных и строчных символов латинского алфавита, а также знаков ! и ?. Длина строки не превышает 100.

Output

Выведите «Panic!» (без кавычек), если сообщение паническое, или «No panic» (без кавычек), если оно не паническое.

Examples

стандартный ввод	стандартный вывод
AAA!!!	Panic!
AAA	No panic
aaa!!!	No panic

Problem J. JokeCoin

Input file: **стандартный ввод**
Output file: **стандартный вывод**
Time limit: 2 секунды
Memory limit: 256 мегабайт

Шахтёр Василий заинтересовался криптовалютой SillyCoin. Майнеры этой валюты получают расписание блоков на следующие сутки. Для каждого блока известен промежуток времени, в течение которого его потребуется майнить, и вознаграждение в монетках SillyCoin, которое гарантированно выдаётся, если этот блок непрерывно майнился в указанное время. Майнер решает сам, стоит ли браться за майнинг того или иного блока. Майнинг следующего блока можно начинать в ту же секунду, как закончена обработка предыдущего блока. У Василия всего одного видеокарта, поэтому он не может майнить несколько блоков одновременно. Кроме того, Василию регулярно приходят счета за электричество, которое придётся оплачивать за фактически потраченное на майнинг время по фиксированному тарифу. Василий хочет узнать, выгодно ли ему заниматься майнингом, поэтому он просит вас написать программу, вычисляющую по заданному расписанию блоков максимально возможную выгоду.

Input

В первой строке находятся два целых числа N и C ($1 \leq N < 86\,400, 0 \leq C \leq 1000$), разделённых пробелом, — количество блоков в расписании и стоимость электричества, затрачиваемого за каждую секунду майнинга, в монетках. В следующих N строках содержится расписание блоков на следующие сутки.

В каждой строке через пробел указаны время начала и окончания блока (в формате **HH:MM:SS** от 00:00:00 до 23:59:59 с разницей минимум в 1 секунду) и вознаграждение P ($0 \leq P \leq 10^5$) в монетках.

Output

Вывести целое число — максимальное вознаграждение, которое Василий может получить за майнинг. Если майнинг принесёт лишь убытки, следует вывести 0.

Examples

стандартный ввод	стандартный вывод
4 0 03:00:00 10:10:00 20 01:00:00 02:30:00 50 16:10:00 19:00:00 100 02:30:00 22:00:00 200	250
3 1 16:59:00 17:00:00 100 01:01:01 01:01:11 20 12:00:00 13:00:00 3601	51
4 10 00:00:05 00:01:55 1100 00:00:10 00:00:21 100 00:01:50 00:02:00 80 23:59:00 23:59:05 40	0

Problem K. King and ICPC

Input file: *стандартный ввод*
Output file: *стандартный вывод*
Time limit: 2 секунды
Memory limit: 256 мебибайт

Тёмные времена нависли над АСМ-ом. Тысячи команд АСМ-щиков вышли на защиту от нападения СТФ-щиков. Всего имеется N команд, пронумерованных от 1 до N . Каждая команда, по традиции, состоит из трёх человек. Каждому человеку присвоили параметр АСМ-овости, выраженный целым числом.

Король АСМ-щиков хочет создать армию, выбрав из каждой команды ровно по одному человеку так, чтобы суммарная АСМ-овость была как можно больше, но при этом делилась нацело на число D .

Так как АСМ-щики по своей натуре любят усложнять задачи, они решили рассмотреть несколько отрезков $[l_i, r_i]$, составляя армию лишь из команд с номерами из этого отрезка.

Король поручил вам обработку этих запросов. На горизонте уже видны флаги, у вас осталось меньше пяти часов!

Input

Первая строка содержит два целых числа N ($1 \leq N \leq 50\,000$) и D ($1 \leq D \leq 50$).

В следующих N строках содержится описание каждой команды в виде трёх целых чисел. АСМ-овость каждого человека является целым числом от 0 до 10^9 .

Следующая строка содержит целое число M — количество отрезков ($1 \leq M \leq 300\,000$).

В следующих M строках идёт описание каждого запроса в виде пары чисел l_i и r_i ($1 \leq l_i \leq r_i \leq N$).

Output

На каждый отрезок выведите в отдельной строке максимальную сумму, которую можно на нём собрать. Если же сумму, делящуюся на D собрать невозможно, выведите «-1» (без кавычек).

Examples

стандартный ввод	стандартный вывод
2 2 0 1 3 1 2 3 3 1 1 2 2 1 2	0 2 6
3 3 0 3 6 1 4 7 1 2 3 4 1 1 1 2 1 3 2 2	6 -1 15 -1

Problem L. Longest Simple Paths

Input file: **стандартный ввод**
Output file: **стандартный вывод**
Time limit: 1 секунда
Memory limit: 256 мегабайт

В связном взвешенном неориентированном графе G *деревом кратчайших путей* называется остовное дерево T графа G , обладающее следующим свойством: путь от корня до любой вершины дерева является кратчайшим путём от корня до этой вершины в исходном графе.

Дерево кратчайших путей можно построить, например, таким способом:

- Выберем вершину в качестве корня, обозначим её за 1.
- Для каждой вершины i в графе G найдём кратчайший путь от корня до вершины i .
- Если кратчайших путей от корня до вершины i более одного, выбирается тот, для которого последовательность номеров вершин лексикографически минимальна.
- Все рёбра в выбранных путях формируют дерево кратчайших путей.

Ваша задача — вычислить максимальную длину простого пути в дереве кратчайших путей, проходящего ровно через k вершин, и сколько различных проходящих через k вершин путей такой длины существует. Два простых пути считаются различными, если множества вершин, через которые они проходят, различны.

Input

Первая строка входных данных содержит три целых числа n, m, k ($1 \leq n \leq 3 \cdot 10^4$, $1 \leq m \leq 6 \cdot 10^4$, $2 \leq k \leq n$) — количество вершин, количество рёбер и количество вершин дерева простых путей в интересующем нас пути, соответственно.

Каждая из последующих строк содержат три целых числа a, b, c ($1 \leq a, b \leq n$, $1 \leq c \leq 10^4$), задающих, что вершины a и b соединены ребром длины c .

Output

Выведите два целых числа — наибольшую длину простого пути, проходящего через k вершин дерева кратчайших путей и количество соответствующих путей такой длины.

Example

стандартный ввод	стандартный вывод
6 6 4 1 2 1 2 3 1 3 4 1 2 5 1 3 6 1 5 6 1	3 4