

Задача А. Словарь

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1.5 секунд
Ограничение по памяти: 256 мегабайт

Дан набор слов и текст, требуется определить для каждого слова, присутствует ли оно в тексте как подстрока.

Формат входных данных

В первой строке дан текст (не более 10^6 строчных латинских букв). Далее дано число M — количество слов в словаре.

В следующих M строках записаны слова (каждое длиной не более 30 строчных латинских букв). Слова различны и отсортированы в лексикографическом порядке.

Суммарная длина слов в словаре не более 10^5 .

Формат выходных данных

M строк вида Yes, если слово присутствует, и No иначе.

Примеры

стандартный ввод	стандартный вывод
trololo	No
3	Yes
abacabadabacaba	Yes
olo	
trol	
abacabadabacabaabacabadabacaba	Yes
1	
abacabadabacabaabacabadabacaba	

Задача В. Библиотека

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Прошел почти год с момента, как Рик оказался на Флорине, однако его сознание никак не прояснялось. Воспоминания о прошлом были спрятаны в глубинах его разума, а может и вовсе утеряны. Однако сегодня что-то случилось. Рик вспомнил: у него была работа. Он анализировал Ничто. Наверное, Ничто — это космос, а значит Рик в прошлом был космоаналитиком. А еще Рик вспомнил, что все жители Флорины должны были погибнуть, но он не знал, почему.

Резидента Мирлина Теренса заинтересовала эта информация, поэтому он взял Рика с собой в библиотеку Верхнего города. Может быть, какая-нибудь литература по космоанализу могла бы вернуть ему память? Теренс не знал, что пропавшего космоаналитика активно ищут, а потому в библиотеке был получен приказ сообщать о любых посетителях, которые спросят о такой литературе. Библиотекарь отследил запросы наших героев в поисковой системе и поспешил вызвать патрульных.

Тем временем Теренс предложил Рiku ознакомиться с книгой известного автора Врийта "Трактат об инструментальном космоанализе". Рiku книга определенно показалась знакомой, особенно его привлекла строка s . Смысла самой строки, он, к сожалению, не понимал, однако в ее частях он видел что-то знакомое. Чтобы разобраться подробнее, Рик решил изучить все подстроки s . Однако изучать равные подстроки не было смысла, а остальные стоило как-либо систематизировать. Например, расставить их по длине и в алфавитном порядке. Поэтому Рик попросил вас узнать, сколько у данной строки существует пар подстрок s_1 и s_2 равной длины, таких, что $s_1 < s_2$ лексикографически.

Формат входных данных

Задана строка s , состоящая из строчных латинских букв ($|s| \leq 2500$).

Формат выходных данных

Выведите одно число — количество искомых пар подстрок.

Пример

стандартный ввод	стандартный вывод
abac	9

Замечание

Рассмотрим подстроки длины 1. Имеется две подстроки "a", каждая из которых меньше подстрок "b" и "c". Также подстрока "b" меньше подстроки "c". Отсюда получаем 5 пар искомых подстрок.

Теперь рассмотрим подстроки длины 2. Подстрока "ab" меньше подстрок "ba" и "ac", а строка "ac" меньше, чем строка "ba". Отсюда получаем еще 3 пары.

Наконец, рассмотрим подстроки длины 3. Подстрока "aba" меньше подстроки "bac".

Таким образом, суммарно получаем 9 искомых пар подстрок.

Задача С. Странный сон

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

Однажды Константин, поучаствовав в очередной, уже 13-й по счету международной олимпиаде, возвращался на поезде домой. Он как всегда сидел и размышлял о смысле жизни, попутно решая задачи по программированию. Через некоторое время Константин задремал, но вот беда, для того, чтобы проснуться, он должен решить всплывшую у него в голове задачу, не дающую ему покоя!

В этот раз Константину приснилось дерево, изначально состоящее всего из одной вершины с номером 1. В поставленной им задаче к дереву постепенно добавлялись новые вершины. В i -ю секунду в дерево добавлялась вершина с номером $i + 1$, которая подвешивалась в качестве сына к вершине p_i , а на ребре между вершинами $i + 1$ и p_i записывалась буква c_i .

Каждому пути из корня дерева до вершины v соответствует некоторая строка, получающаяся путем выписывания символов, записанных на ребрах текущего пути в порядке следования от корня к вершине v . Перед Константином стояла нелегкая на первый взгляд задача — после каждого добавления новой вершины посчитать количество уникальных строк, начинающихся в корне дерева (вершине с номером 1), и заканчивающихся в какой-либо другой вершине.

В своем сне Константин вовсе не гений, поэтому решить эту задачу сам он не в силах. Помогите Константину решить задачу и тем самым проснуться.

Формат входных данных

В первой строке записано число n — количество запросов на добавление новой вершины в дерево ($1 \leq n \leq 3 \cdot 10^5$).

В следующих n строках описаны запросы добавления вершин. i -й запрос описывается параметрами p_i ($1 \leq p_i \leq i$) и c_i , которые означают, что добавленная вершина с номером $i + 1$ подвешивается к вершине с номером p_i в качестве потомка, а на полученном ребре записывается символ c_i — строчная буква латинского алфавита.

Формат выходных данных

Выведите n строк. В i -й строке выведите ответ на задачу Константина после добавления $(i + 1)$ -й вершины.

Примеры

стандартный ввод	стандартный вывод
2 1 b 2 p	1 2
3 1 o 1 o 2 j	1 1 2

Задача D. К-я строка

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

Реализуйте структуру данных, которая поддерживает следующие операции:

- добавить в словарь строку S ;
- найти в словаре k -ю строку в лексикографическом порядке.

Изначально словарь пуст.

Формат входных данных

Первая строка входного файла содержит число N — количество команд ($1 \leq N \leq 10^5$).

Последующие N строк содержат по одной команде каждая. Команды записываются следующим образом:

- 1 S — добавить строку S в словарь;
- 2 k — вывести k -ю строку в лексикографическом порядке.

Гарантируется, что при запросе k -й строки она существует. Также гарантируется, что сумма длин всех добавляемых строк не превышает $3 \cdot 10^5$.

Все строки состоят из строчных латинских букв.

Формат выходных данных

Для каждого запроса второго типа выведите k -ю в лексикографическом порядке строчку из словаря на момент запроса.

Гарантируется, что суммарная длина строк в выходном файле не превышает 10^5 .

Пример

стандартный ввод	стандартный вывод
7	tolstoy
1 pushkin	gogol
1 lermontov	
1 tolstoy	
1 gogol	
1 gorkiy	
2 5	
2 1	

Задача Е. Мультимножество Василия

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

У автора уже закончились истории про Василия, поэтому он просто написал формальную постановку задачи.

У вас есть q запросов и мультимножество A , изначально содержащее только число 0. Запросы бывают трёх видов:

- «+ x » — добавить в мультимножество A число x .
- «- x » — удалить одно вхождение числа x из мультимножества A . Гарантируется, что хотя бы одно число x в этот момент присутствует в мультимножестве.
- «? x » — вам даётся число x , требуется вычислить $\max_{y \in A} x \oplus y$, то есть максимальное значение побитового исключающего ИЛИ (также известно как XOR) числа x и какого-нибудь числа y из мультимножества A .

Мультимножество — это множество, в котором разрешается несколько одинаковых элементов.

Формат входных данных

В первой строке входных данных содержится число q ($1 \leq q \leq 200\,000$) — количество запросов, которые требуется обработать Василию.

Каждая из последующих q строк входных данных содержит один трёх символов «+», «-» или «?» и число x_i ($1 \leq x_i \leq 10^9$). Гарантируется, что во входных данных встречается хотя бы один запрос «?».

Обратите внимание, что число 0 всегда будет присутствовать в мультимножестве.

Формат выходных данных

На каждый запрос типа «?» выведите единственное целое число — максимальное значение побитового исключающего ИЛИ для числа x_i и какого-либо числа из мультимножества A .

Пример

стандартный ввод	стандартный вывод
10	11
+ 8	10
+ 9	14
+ 11	13
+ 6	
+ 1	
? 3	
- 8	
? 3	
? 8	
? 11	

Замечание

После первых пяти операций в мультимножестве A содержатся числа 0, 8, 9, 11, 6 и 1.

Ответом на шестой запрос будет число $11 = 3 \oplus 8$ максимальное из чисел $3 \oplus 0 = 3$, $3 \oplus 9 = 10$, $3 \oplus 11 = 8$, $3 \oplus 6 = 5$ и $3 \oplus 1 = 2$.

Задача F. Фенвик с подвохом

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.4 секунд
Ограничение по памяти:	256 мегабайт

Вам дан массив из n элементов, изначально заполненный нулями. Вам поступают запросы двух типов:

- прибавить ко всем числам на $[L : R]$ число x
- узнать значение i -го элемента.

Реализуйте дерево Фенвика, которое поддерживает данные операции. Нет, мы не ошиблись. Да, это дерево Фенвика. Да, оно на это способно (а мы верим, что вы способны это придумать). Нет, слишком долго работать не будет. И да, дерево отрезков здесь не зайдет, даже не пытайтесь.

Формат входных данных

В первой строке даны два числа N и M — количество элементов массива и количество запросов соответственно ($1 \leq N \leq 2 \cdot 10^5, 1 \leq M \leq 6 \cdot 10^5$)

В следующих m строках описаны запросы:

Запросы первого типа имеют вид $+LRx$ ($1 \leq L, R \leq N, -100 \leq x \leq 100$).

Запросы второго типа имеют вид $?i$ ($1 \leq i \leq N$)

Формат выходных данных

На каждый запрос второго типа выведите элемент стоящий на i -й позиции массива.

Пример

стандартный ввод	стандартный вывод
5 5	7
+ 1 5 7	10
? 3	7
+ 1 3 3	
? 2	
? 4	

Задача G. 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 Н. Разреженные таблицы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

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 входного файла даны 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.

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_m, v_m и ans_m (последний запрос и ответ на него).

Пример

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

Задача I. 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 5000000003 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.