

Задача А. Разреженные таблицы

Имя входного файла: `sparse.in`
Имя выходного файла: `sparse.out`
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 входного файла

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 ($0 \leq a_1 < 16714589$) — 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 выходного файла

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

Примеры

<code>sparse.in</code>	<code>sparse.out</code>
10 8 12345	5 3 1565158
3 9	

Задача В. Вперёд!

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

Капрал Дукар любит раздавать приказы своей роте. Самый любимый его приказ — «Вперёд!». Капрал строит солдат в ряд и отдаёт некоторое количество приказов, каждый из которых звучит так: «Рядовые с l_i по l_j — вперёд!»

Перед тем, как Дукар отдал первый приказ, солдаты были пронумерованы от 1 до n слева направо. Услышав приказ «Рядовые с l_i по l_j — вперёд!», солдаты, стоящие на местах с l_i по l_j включительно, продвигаются в начало ряда в том же порядке, в котором были.

Например, если в какой-то момент солдаты стоят в порядке 2, 3, 6, 1, 5, 4, то после приказа «Рядовые с 2 по 4 — вперёд!», порядок будет таким: 3, 6, 1, 2, 5, 4. А если потом Капрал вышлет вперёд солдат с 3 по 4, то порядок будет уже таким: 1, 2, 3, 6, 5, 4.

Вам дана последовательность приказов Капрала. Найдите порядок, в котором будут стоять солдаты после исполнения всех приказов.

Формат входного файла

В первой строке входного файла указаны числа n и m ($2 \leq n \leq 100\,000$, $1 \leq m \leq 100\,000$) — число солдат и число приказов. Следующие m строк содержат приказы в виде двух целых чисел: l_i и r_i ($1 \leq l_i \leq r_i \leq n$).

Формат выходного файла

Выведите в выходной файл n целых чисел — порядок, в котором будут стоять солдаты после исполнения всех приказов.

Примеры

<code>movetofront.in</code>	<code>movetofront.out</code>
6 3 2 4 3 5 2 2	1 4 5 2 3 6

Задача С. Шаги

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

Саша и Вася начали заниматься чечеткой. Этот танец состоит из притопывания ног по полу. Но так как они очень быстро учатся, они решили поэкспериментировать с хореографией.

Хореография чечетки описывается последовательностью из двух букв 'L' и 'R'. 'L' означает притопывание левой ногой, а 'R', соответственно, правой. Саша понял, что самая лучшая часть этого танца это та, в которой не используется одна и та же нога подряд. Он определил значение хореографии как наибольшую непрерывную последовательность, в которой нет двух одинаковых последовательных символов.

Как известно, создание красивого танца – очень непростой процесс, с множеством маленьких изменений, прежде чем оптимальный вариант будет найден. Поэтому Вася хочет знать значение хореографии после каждого изменения. Изменение – это замена L на R (или наоборот) в какой-то позиции.

До всех изменений последовательность состоит только из одних букв L.

Формат входного файла

На первой строке задано количество два целых числа: длина хореографии N ($1 \leq N \leq 200000$) и число изменений Q ($1 \leq Q \leq 200000$).

Следующие Q строк содержат по одному числу, определяющего в какой позиции происходит замена.

Формат выходного файла

Вы должны вывести Q чисел, по одному на каждой строке: значения хореографии после каждого изменения.

Примеры

<code>step.in</code>	<code>step.out</code>
6 2	3
2	5
4	
6 5	3
4	3
1	3
1	5
2	6
6	