

Задача А. Снеговики

Имя входного файла: `snowmen.in`
Имя выходного файла: `snowmen.out`
Ограничение по времени: 4 секунды
Ограничение по памяти: 64 мегабайта

Зима. 2012 год. На фоне грядущего Апокалипсиса и конца света незамеченной прошла новость об очередном прорыве в областях клонирования и снеговиков: клонирования снеговиков. Вы конечно знаете, но мы вам напомним, что снеговик состоит из нуля или более вертикально поставленных друг на друга шаров, а клонирование — это процесс создания идентичной копии (клона).

В местечке Местячково учитель Андрей Сергеевич Учитель купил через интернет-магазин «Интернет-магазин аппаратов клонирования» аппарат для клонирования снеговиков. Теперь дети могут играть и даже играют во дворе в следующую игру. Время от времени один из них выбирает понравившегося снеговика, клонирует его и:

- либо добавляет ему сверху один шар;
- либо удаляет из него верхний шар (если снеговик не пустой).

Учитель Андрей Сергеевич Учитель записал последовательность действий и теперь хочет узнать суммарную массу всех построенных снеговиков.

Формат входных данных

Первая строка содержит количество действий n ($1 \leq n \leq 200\,000$). В строке номер $i + 1$ содержится описание действия i :

- $t\ m$ — клонировать снеговика номер t ($0 \leq t < i$) и добавить сверху шар массой m ($0 < m \leq 1000$);
- $t\ \emptyset$ — клонировать снеговика номер t ($0 \leq t < i$) и удалить верхний шар. Гарантируется, что снеговик t не пустой.

В результате действия i , описанного в строке $i + 1$ создается снеговик номер i . Изначально имеется пустой снеговик с номером ноль.

Все числа во входном файле целые.

Формат выходных данных

Выведите суммарную массу построенных снеговиков.

Примеры

<code>snowmen.in</code>	<code>snowmen.out</code>
8	74
0 1	
1 5	
2 4	
3 2	
4 3	
5 0	
6 6	
1 0	

Задача В. Откат

Имя входного файла: `rollback.in`
Имя выходного файла: `rollback.out`
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайта

Сергей работает системным администратором в очень крупной компании. Естественно, в круг его обязанностей входит резервное копирование информации, хранящейся на различных серверах и «откат» к предыдущей версии в случае возникновения проблем.

В данный момент Сергей борется с проблемой недостатка места для хранения информации для восстановления. Он решил перенести часть информации на новые сервера. К сожалению, если что-то случится во время переноса, он не сможет произвести откат, поэтому процедура переноса должна быть тщательно спланирована.

На данный момент у Сергея хранятся n точек восстановления различных серверов, пронумерованных от 1 до n . Точка восстановления с номером i позволяет произвести откат для сервера a_i . Сергей решил разбить перенос на этапы, при этом на каждом этапе в случае возникновения проблем будут доступны точки восстановления с номерами $l, l + 1, \dots, r$ для некоторых l и r .

Для того, чтобы спланировать перенос данных оптимальным образом, Сергею необходимо научиться отвечать на запросы: для заданного l , при каком минимальном r в процессе переноса будут доступны точки восстановления не менее чем k различных серверов.

Помогите Сергею.

Формат входных данных

Первая строка входного файла содержит два целых числа n и m , разделенные пробелами — количество точек восстановления и количество серверов ($1 \leq n, m \leq 100\,000$). Вторая строка содержит n целых чисел a_1, a_2, \dots, a_n — номера серверов, которым соответствуют точки восстановления ($1 \leq a_i \leq m$).

Третья строка входного файла содержит q — количество запросов, которые необходимо обработать ($1 \leq q \leq 100\,000$). В процессе обработки запросов необходимо поддерживать число p , исходно оно равно 0. Каждый запрос задается парой чисел x_i и y_i , используйте их для получения данных запроса следующим образом: $l_i = ((x_i + p) \bmod n) + 1$, $k_i = ((y_i + p) \bmod m) + 1$ ($1 \leq l_i, x_i \leq n$, $1 \leq k_i, y_i \leq m$). Пусть ответ на i -й запрос равен r . После выполнения этого запроса, следует присвоить p значение r .

Формат выходных данных

На каждый запрос выведите одно число — искомое минимальное r , либо 0, если такого r не существует.

Примеры

<code>rollback.in</code>	<code>rollback.out</code>
7 3	1
1 2 1 3 1 2 1	4
4	0
7 3	6
7 1	
7 1	
2 2	

Задача С. Урны и шары

Имя входного файла: `balls.in`
Имя выходного файла: `balls.out`
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайт

Пусть у вас есть n урн, в каждой из которых лежит по одному шару. Урна с номером i содержит шарик под номером i . У вас есть специальное устройство, которое позволяет перемещать шарики. Им чрезвычайно просто пользоваться: сначала вы выбираете некоторый отрезок последовательных урн. После этого вы выбираете некоторый другой отрезок последовательных урн такой же длины, как и исходный, и затем шарики из урн первого отрезка перемещаются в соответствующие урны второго отрезка.

Дана последовательность перемещений. Установите, в какой урне окажется каждый шарик.

Формат входных данных

Первая строка входных данных содержит два числа n и m — число урн и число перемещений, соответственно ($1 \leq n \leq 100\,000$, $1 \leq m \leq 50\,000$). Каждая из следующих m строк содержит три числа $count_i$, $from_i$ и to_i , которые означают одновременное перемещение всех шариков из урны $from_i$ в урну to_i , всех шариков из урны $from_i + 1$ в урну $to_i + 1$, ..., всех шариков из урны $from_i + count_i - 1$ в урну $to_i + count_i - 1$ ($1 \leq count_i, from_i, to_i \leq n$, $\max(from_i, to_i) + count_i \leq n + 1$).

Формат выходных данных

Выведите n чисел — итоговые позиции каждого шарика.

Примеры

<code>balls.in</code>	<code>balls.out</code>
2 3	1 1
1 1 2	
1 2 1	
1 2 1	

Задача D. Океанский бой

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

В игре «Океанский бой», которая проходит на поле $M \times N$, первым этапом является расстановка кораблей. Изначально поле является пустым, а в качестве кораблей могут использоваться произвольные прямоугольники, состоящие из клеток поля. При этом никакие два корабля не могут иметь общую точку.

Ценность корабля равна количеству клеток, которые он занимает.

По заданной частичной расстановке кораблей (несколько кораблей, поставленных в соответствии с правилами) выясните, какова может быть максимальная ценность одного корабля в финальной расстановке, если разрешается поставить дополнительно любое количество кораблей в соответствии с правилами.

Формат входных данных

В первой строке входа заданы два целых числа M и N — размеры поля ($1 \leq M, N \leq 5000$). Следующая строка содержит целое число K ($0 \leq K \leq 10^4$) — количество расставленных кораблей. Каждая из следующих K строк содержит по четыре целых числа X_1, Y_1, X_2, Y_2 — координаты противоположных вершин корабля ($0 \leq X_1, X_2 \leq M, 0 \leq Y_1, Y_2 \leq N, X_1 \neq X_2, Y_1 \neq Y_2$). Гарантируется, что заданная на входе расстановка корректна, то есть никакие два поставленных корабля не могут иметь общую точку.

Формат выходных данных

Выведите одно целое число — максимальную ценность корабля в расстановке, которая включает в себя все заданные корабли на заданных местах.

Примеры

стандартный ввод	стандартный вывод
4 4 1 0 0 1 2	8
7 3 2 1 3 2 0 6 0 7 1	6

Задача Е. Суровый корректор

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

По мнению Александра Павловича, текст необычайно красив, если некоторые особые слова (например, «коммунизм», «Ленин», «счастье») встречаются не слишком часто, но и не слишком редко, к тому же достаточно равномерно. Александр Павлович работает корректором. К нему поступают тексты, он имеет право их некоторым образом менять, после чего возвращает уже исправленную версию. В связи со своими воззрениями о красоте Александру Павловичу постоянно приходится проверять, сколько особых слов сейчас в той или иной части текста. Он настолько устал от рутинного подсчёта: «а сколько тут особых слов?», «а сколько тут?», что просит вас помочь ему автоматизировать этот процесс.

Формат входных данных

Первая строка входного файла содержит текст, в котором ищутся особые слова. Следующая строка содержит N ($1 \leq N \leq 10^5$) — количество особых слов. Следующие N строк содержат особые слова. Все особые слова различны. Суммарная длина строк не превосходит 10^5 . В следующей строке дано Q ($1 \leq Q \leq 10^5$) — количество интересных Александру Павловичу отрезков. Следующие Q строк содержат сами отрезки.

Формат выходных данных

Выведите Q чисел — количества вхождений особых слов в соответствующий отрезок текста.

Примеры

<code>corrector.in</code>	<code>corrector.out</code>
abacababa	5 0 2
2	
a	
aba	
3	
5 9	
2 2	
2 6	