

## Задача А. Ромашки

Имя входного файла: `camomiles.in`  
Имя выходного файла: `camomiles.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Саша любит наблюдать за ромашками. Но следить за всем полем сразу ей тяжело. Поэтому она наблюдает только за его частью, ограниченной прямоугольником размером  $x \times y$ . Этот прямоугольник поделен на маленькие прямоугольники размером  $1 \times 1$ . Во время ее наблюдений могут происходить следующие события:

1. в каком-то прямоугольнике появляются или исчезают несколько ромашек;
2. к ней может заглянуть ее подруга Настя и поинтересоваться, сколько видно ромашек в части поля, состоящей из нескольких прямоугольников.

### Формат входного файла

Первая строка входного файла содержит два натуральных числа  $x$  и  $y$  ( $1 \leq x, y \leq 1000$ ). Следующая строка содержит единственное натуральное число  $n$  ( $1 \leq n \leq 100\,000$ ) — количество произошедших событий. Далее следуют  $n$  записей о происшедших событиях по одной в строке. В начале строки записано число  $t$ . Если  $t$  равно:

- 1, то за ним следуют 3 числа:  $x_1, y_1$  ( $1 \leq x_1 \leq x, 1 \leq y_1 \leq y$ ) и  $k$  ( $-10\,000 \leq k \leq 10\,000$ ) — координаты прямоугольника и величина, на которую в нем изменилось количество видимых ромашек;
- 2, то за ним следуют 4 числа:  $x_1, y_1, x_2, y_2$  ( $1 \leq x_1 \leq x_2 \leq x, 1 \leq y_1 \leq y_2 \leq y$ ), которые означают, что Настя попросила подсчитать количество ромашек в прямоугольниках  $(x', y')$  из области:  $x_1 \leq x' \leq x_2, y_1 \leq y' \leq y_2$ .

### Формат выходного файла

Для каждого Настиного вопроса выведите искомое количество ромашек (оно может быть отрицательным).

### Примеры

<code>camomiles.in</code>	<code>camomiles.out</code>
8 8	3
3	
1 2 2 2	
1 1 1 1	
2 1 1 2 2	

## Задача В. Следующий

Имя входного файла: `next.in`  
Имя выходного файла: `next.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Реализуйте структуру данных, которая поддерживает множество  $S$  целых чисел, с которым разрешается производить следующие операции:

- $add(i)$  — добавить в множество  $S$  число  $i$  (если он там уже есть, то множество не меняется);
- $next(i)$  — вывести минимальный элемент множества, не меньший  $i$ . Если искомый элемент в структуре отсутствует, необходимо вывести  $-1$ .

### Формат входного файла

Исходно множество  $S$  пусто. Первая строка входного файла содержит  $n$  — количество операций ( $1 \leq n \leq 300\,000$ ). Следующие  $n$  строк содержат операции. Каждая операция имеет вид либо «+  $i$ », либо «?  $i$ ». Операция «?  $i$ » задает запрос  $next(i)$ .

Если операция «+  $i$ » идет во входном файле в начале или после другой операции «+», то она задает операцию  $add(i)$ . Если же она идет после запроса «?», и результат этого запроса был  $y$ , то выполняется операция  $add((i + y) \bmod 10^9)$ .

Во всех запросах и операциях добавления параметры лежат в интервале от 0 до  $10^9$ .

### Формат выходного файла

Для каждого запроса выведите одно число — ответ на запрос.

### Пример

<code>next.in</code>	<code>next.out</code>
6	3
+ 1	4
+ 3	
+ 3	
? 2	
+ 1	
? 4	

## Задача С. В начало строя!

Имя входного файла: movetofront.in  
Имя выходного файла: movetofront.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Капрал Питуца любит командовать своим отрядом. Его любимый приказ «в начало строя». Он выстраивает свой отряд в шеренгу и оглашает последовательность приказов. Каждая приказ имеет вид «Солдаты с  $l_i$  по  $r_i$  — в начало строя!»

Пронумеруем солдат в начальном положении с 1 до  $n$ , слева направо. Приказ «Солдаты с  $l_i$  по  $r_i$  — в начало строя!» означает, что солдаты, стоящие с  $l_i$  по  $r_i$  включительно перемещаются в начало строя, сохраняя относительный порядок.

Например, если в некоторый момент солдаты стоят в порядке 2, 3, 6, 1, 5, 4, после приказа: «Солдаты с 2 по 4 — в начало строя!» порядок будет 3, 6, 1, 2, 5, 4.

По данной последовательности приказов найти конечный порядок солдат в строю.

### Формат входного файла

В первой строке два целых числа  $n$  and  $m$  ( $2 \leq n \leq 100\,000$ ,  $1 \leq m \leq 100\,000$ ) — количество солдат и количество приказов. Следующие  $m$  строк содержат по два целых числа  $l_i$  и  $r_i$  ( $1 \leq l_i \leq r_i \leq n$ ).

### Формат выходного файла

Выведите  $n$  целых чисел — порядок солдат в конечном положении после выполнения всех приказов.

### Пример

movetofront.in	movetofront.out
6 3	1 4 5 2 3 6
2 4	
3 5	
2 2	