

## Задача A. RSQ

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

### Формат входных данных

В первой строке находится число  $n$  — размер массива ( $1 \leq n \leq 500\,000$ ). Во второй строке находится  $n$  чисел  $a_i$  — элементы массива. Далее содержится описание операций, их количество не превышает  $1\,000\,000$ . В каждой строке находится одна из следующих операций:

- `set i x` — установить  $a[i]$  в  $x$ .
- `sum i j` — вывести значение суммы элементов в массиве на отрезке с  $i$  по  $j$ , гарантируется, что  $(1 \leq i \leq j \leq n)$ .

Все числа во входном файле и результаты выполнения всех операций не превышают по модулю  $10^{18}$ .

### Формат выходных данных

Выведите последовательно результат выполнения всех операций `sum`. Следуйте формату выходного файла из примера.

### Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 5                | 14                |
| 1 2 3 4 5        | 15                |
| sum 2 5          | 10                |
| sum 1 5          | 9                 |
| sum 1 4          | 12                |
| sum 2 4          | 22                |
| set 1 10         | 20                |
| set 2 3          | 10                |
| set 5 2          |                   |
| sum 2 5          |                   |
| sum 1 5          |                   |
| sum 1 4          |                   |
| sum 2 4          |                   |

# Задача В. RMQ2

Имя входного файла:            стандартный ввод

Имя выходного файла:           стандартный вывод

Ограничение по времени:        2 секунды

Ограничение по памяти:          256 мегабайт

## Формат входных данных

В первой строке находится число  $n$  — размер массива ( $1 \leq n \leq 10^5$ ). Во второй строке находится  $n$  чисел  $a_i$  — элементы массива. Далее содержится описание операций, их количество не превышает  $2 \cdot 10^5$ . В каждой строке находится одна из следующих операций:

- `set i j x` — установить все  $a[k]$ ,  $i \leq k \leq j$  в  $x$ .
- `add i j x` — увеличить все  $a[k]$ ,  $i \leq k \leq j$  на  $x$ .
- `min i j` — вывести значение минимального элемента в массиве на отрезке с  $i$  по  $j$ , гарантируется, что  $(1 \leq i \leq j \leq n)$ .

Все числа во входном файле и результаты выполнения всех операций не превышают по модулю  $10^{18}$ .

## Формат выходных данных

Выведите последовательно результат выполнения всех операций `min`. Следуйте формату выходного файла из примера.

## Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 5                | 2                 |
| 1 2 3 4 5        | 1                 |
| min 2 5          | 1                 |
| min 1 5          | 2                 |
| min 1 4          | 5                 |
| min 2 4          | 5                 |
| set 1 3 10       | 8                 |
| add 2 4 4        | 8                 |
| min 2 5          |                   |
| min 1 5          |                   |
| min 1 4          |                   |
| min 2 4          |                   |

## Задача С. Окна

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

На экране расположены прямоугольные окна, каким-то образом перекрывающиеся (со сторонами, параллельными осям координат). Вам необходимо найти точку, которая покрыта наибольшим числом из них.

### Формат входных данных

В первой строке входного файла записано число окон  $n$  ( $1 \leq n \leq 50000$ ). Следующие  $n$  строк содержат координаты окон  $x_{(1,i)} y_{(1,i)} x_{(2,i)} y_{(2,i)}$ , где  $(x_{(1,i)}, y_{(1,i)})$  — координаты левого верхнего угла  $i$ -го окна, а  $(x_{(2,i)}, y_{(2,i)})$  — правого нижнего (на экране компьютера  $y$  растет сверху вниз, а  $x$  — слева направо). Все координаты — целые числа, по модулю не превосходящие  $2 \cdot 10^5$ .

### Формат выходных данных

В первой строке выходного файла выведите максимальное число окон, покрывающих какую-либо из точек в данной конфигурации. Во второй строке выведите два целых числа, разделенные пробелом — координаты точки, покрытой максимальным числом окон. Окна считаются замкнутыми, т.е. покрывающими свои граничные точки.

### Примеры

| стандартный ввод        | стандартный вывод |
|-------------------------|-------------------|
| 2<br>0 0 3 3<br>1 1 4 4 | 2<br>1 3          |
| 1<br>0 0 1 1            | 1<br>0 1          |

Задача D. Различные числа

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

Сколько различных чисел на отрезке массива?

Формат входных данных

На первой строке длина массива  $n$  ( $1 \leq n \leq 300\,000$ ). На второй строке  $n$  целых чисел от 0 до  $10^9$ . На третьей строке количество запросов  $q$  ( $1 \leq q \leq 300\,000$ ). Следующие  $q$  строк содержат описание запросов, по одному на строке. Каждый запрос задаётся парой целых чисел  $l, r$  ( $1 \leq l \leq r \leq n$ ).

Формат выходных данных

Выведите ответы на запросы по одному в строке.

Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 5                | 3                 |
| 1 1 2 1 3        | 2                 |
| 3                | 3                 |
| 1 5              |                   |
| 2 4              |                   |
| 3 5              |                   |

## Задача Е. Художник

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

Итальянский художник-абстракционист Ф. Мандарино увлекся рисованием одномерных черно-белых картин. Он пытается найти оптимальное местоположение и количество черных участков картины. Для этого он проводит на прямой белые и черные отрезки, и после каждой из таких операций хочет знать количество черных отрезков на получившейся картине и их суммарную длину.

Изначально прямая — белая. Ваша задача — написать программу, которая после каждой из таких операций выводит в выходной файл интересующие художника данные.

### Формат входных данных

В первой строке входного файла содержится общее количество нарисованных отрезков ( $1 \leq n \leq 100\,000$ ). В последующих  $n$  строках содержится описание операций. Каждая операция описывается строкой вида  $c\ x\ l$ , где  $c$  — цвет отрезка (W для белых отрезков, B для черных), а сам отрезок имеет вид  $[x; x + l)$ , причем координаты обоих концов — целые числа, не превосходящие по модулю 500 000. Длина задается положительным целым числом.

### Формат выходных данных

После выполнения каждой из операций необходимо вывести в выходной файл на отдельной строке количество черных отрезков на картине и их суммарную длину, разделенные одним пробелом.

### Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 7                | 0 0               |
| W 2 3            | 1 2               |
| B 2 2            | 1 4               |
| B 4 2            | 1 4               |
| B 3 2            | 2 6               |
| B 7 2            | 3 5               |
| W 3 1            | 0 0               |
| W 0 10           |                   |

## Задача F. Deda

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

Маленькая Марики сочиняет бессмысленную необычную сказку и рассказывает своему деду, который продолжает ее перебивать и задавать ей глупые интригующие вопросы. В сказке Марики  $N$  детей, обозначенных номерами от 1 до  $N$  по возрасту (от младшего, обозначенного 1, до самого старшего, обозначенного  $N$ ), отправились в поездку на поезде. Поезд отходит от станции 0 и останавливается по порядку на станциях 1, 2, 3... до бесконечности. Каждое из следующих утверждений Марики имеет форму: «На остановке  $X$  ребенок  $A$  вышел», где порядок этих утверждений совершенно произвольный. Другими словами, это не зависит от порядка станции. Ее дедушка иногда задает вопрос в форме: «Исходя из приведенных выше утверждений, кто младший из высадившихся детей, обозначенных числом, большим или равным  $B$  и которые доехали до  $Y$  или меньше?» Если на данный момент дедушка задает вопрос, но до сих пор не было сказано, что ребенок выходит из поезда, мы предполагаем, что ребенок едет бесконечно много остановок. Марики должна дать правильный ответ на каждый из вопросов дедушки, иначе дедушка разозлится и уснет. Ответ должен быть правильным в тот момент, когда дед задает вопрос, хотя он может измениться позже, учитывая новые заявления Марики, но это не имеет значения. Напишите программу, которая отслеживает заявления Марики и отвечает на вопросы ее деда.

### Формат входных данных

Первая строка содержит положительные целые числа  $N$  и  $Q$  ( $2 \leq N, Q \leq 200000$ ), количество детей и количество заявлений. Каждая из следующих  $Q$  строк описывает:

- либо утверждение Марики в форме «M»  $X$   $A$ , где «M» обозначает Марику, а  $X$  и  $A$  — положительные целые числа ( $1 \leq X \leq 1000000000$ ,  $1 \leq A \leq N$ ) из задачи,
- или вопрос ее дедушки в форме «D»  $Y$   $B$ , где «D» обозначает дедушку, а  $Y$  и  $B$  — положительные целые числа ( $1 \leq Y \leq 1000000000$ ,  $1 \leq B \leq N$ ) из задачи.

Все высказывания Марики соответствуют разным детям, и по крайней мере одна строка во вопрос ее дедушки.

### Формат выходных данных

Для вопроса каждого дедушки выведите номер нужного ребенка в отдельной строке. Если такого ребёнка не существует, вывод -1.

## Примеры

| стандартный ввод   | стандартный вывод       |
|--|-------------------------|
| 3 4<br>M 10 3<br>M 5 1<br>D 20 2<br>D 5 1  | 3<br>1                  |
| 10 10<br>M 20 10<br>D 1 9<br>M 2 3<br>D 17 10<br>M 20 2<br>D 8 2<br>M 40 1<br>D 25 2<br>M 33 9<br>D 37 9 | -1<br>-1<br>3<br>2<br>9 |

## Задача G. И снова сумма

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 5 секунд  
Ограничение по памяти: 512 мегабайт

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

- $\text{add}(i)$  — добавить в множество  $S$  число  $i$  (если оно там уже есть, то множество не меняется);
- $\text{sum}(l, r)$  — вывести сумму всех элементов  $x$  из  $S$ , которые удовлетворяют неравенству  $l \leq x \leq r$ .

Исходно множество  $S$  пусто.

### Формат входных данных

Первая строка содержит  $n$  — количество операций ( $1 \leq n \leq 300\,000$ ). Следующие  $n$  строк содержат операции. Каждая операция имеет вид либо «+  $i$ », либо «?  $l$   $r$ ». Операция «?  $l$   $r$ » задает запрос  $\text{sum}(l, r)$ .

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

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

### Формат выходных данных

Для каждого запроса выведите одно число — ответ на запрос.

### Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 6                | 3                 |
| + 1              | 7                 |
| + 3              |                   |
| + 3              |                   |
| ? 2 4            |                   |
| + 1              |                   |
| ? 2 4            |                   |