

Задача А. Сумма

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

Дан массив из N элементов, нужно научиться находить сумму чисел на отрезке.

Формат входного файла

Первая строка входного файла содержит два целых числа N и K — число чисел в массиве и количество запросов. ($1 \leq N \leq 100\,000$), ($0 \leq K \leq 100\,000$). Следующие K строк содержат запросы:

1. `A l r x` — присвоить элементам массива с позициями от l до r значение x ($1 \leq l \leq r \leq N$, $0 \leq x \leq 10^9$)
2. `Q l r` — найти сумму чисел в массиве на позициях от l до r . ($1 \leq l \leq r \leq N$)

Изначально массив заполнен нулями.

Формат выходного файла

На каждый запрос вида `Q l r` нужно вывести единственное число — сумму на отрезке.

Примеры

| sum.in | sum.out |
|---------|---------|
| 5 9 | 3 |
| A 2 3 2 | 2 |
| A 3 5 1 | 3 |
| A 4 5 2 | 4 |
| Q 1 3 | 2 |
| Q 2 2 | 7 |
| Q 3 4 | |
| Q 4 5 | |
| Q 5 5 | |
| Q 1 5 | |

Задача В. Художник

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

Не успев дорисовать свой гениальный футуристический шедевр, М. Калевич 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 такой операции выводит в выходной файл интересные художника данные.

Формат входного файла

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 и 'B' для черных), а сам отрезок имеет вид $[x; x + l]$, причем координаты обоих концов — целые числа, по модулю не превосходящие 500 000. Длина задается положительным целым числом.

Формат выходного файла

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.

Примеры

| painter.in | painter.out |
|------------|-------------|
| 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 С. Вперед!

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 них 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 из 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

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

Примечание**Задача D. Переворот**

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

Дан массив. Надо научиться обрабатывать два типа запросов.

- 1 L R - перевернуть отрезок [L, R]
- 2 L R - найти минимум на отрезке [L, R]

Формат входного файла

Первая строка файла содержит два числа n , m . ($1 \leq n, m \leq 10^5$) Во второй строке находится n чисел a_i ($1 \leq a_i \leq 10^9$) - исходный массив. Остальные m строк содержат запросы, в формате описанном в условии. Для чисел L,R выполняется ограничение ($1 \leq L \leq R \leq n$).

Формат выходного файла

На каждый запрос типа 2, во входной файл выведите ответ на него, в отдельной строке.

Примеры

| reverse.in | reverse.out |
|------------------------|-------------|
| 10 7 | 3 |
| 5 3 2 3 12 6 7 5 10 12 | 2 |
| 2 4 9 | 2 |
| 1 4 6 | 2 |
| 2 1 8 | |
| 1 1 8 | |
| 1 8 9 | |
| 2 1 7 | |
| 2 3 6 | |