

Задача А. RMQ

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

Формат входного файла

В первой строке находится число n — размер массива. ($1 \leq n \leq 500000$) Во второй строке находится n чисел a_i — элементы массива. Далее содержится описание операций, их количество не превышает 1000000. В каждой строке находится одна из следующих операций:

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

В массив помещаются только целые числа, не превышающие по модулю 10^9 .

Формат выходного файла

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

Пример

<code>rmq.in</code>	<code>rmq.out</code>
5	2
1 2 3 4 5	1
min 2 5	1
min 1 5	2
min 1 4	2
min 2 4	2
set 1 10	3
set 2 3	3
set 5 2	
min 2 5	
min 1 5	
min 1 4	
min 2 4	

Задача В. RSQ

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

Формат входного файла

В первой строке находится число n — размер массива. ($1 \leq n \leq 500000$) Во второй строке находится n чисел a_i — элементы массива. Далее содержится описание операций, их количество не превышает 1000000. 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 выходного файла

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

Пример

rsq.in	rsq.out
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 С. Криптография

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

Задано n матриц A_1, A_2, \dots, A_n размера 2×2 . Необходимо для нескольких запросов вычислить произведение матриц A_i, A_{i+1}, \dots, A_j . Все вычисления производятся по модулю r .

Формат входного файла

Первая строка входного файла содержит числа r ($1 \leq r \leq 10\,000$), n ($1 \leq n \leq 200\,000$) и m ($1 \leq m \leq 200\,000$). Следующие n блоков по две строки содержащие по два числа в строке — описания матриц. Затем следуют m пар целых чисел от 1 до n , запросы на произведение на отрезке.

Формат выходного файла

Выведите m блоков по две строки, по два числа в каждой — произведения на отрезках. Разделяйте блоки пустой строкой. Все вычисления производятся по модулю r .

Пример

<code>crypto.in</code>	<code>crypto.out</code>
3 4 4	0 2
0 1	0 0
0 0	
	0 2
2 1	0 1
1 2	
	0 1
0 0	0 0
0 2	
	2 1
1 0	1 2
0 2	
1 4	
2 3	
1 3	
2 2	

Задача D. RMQ

Имя входного файла: `rmq2.in`
Имя выходного файла: `rmq2.out`

Формат входного файла

В первой строке находится число n — размер массива. ($1 \leq n \leq 100000$) Во второй строке находится n чисел a_i — элементы массива. Далее содержится описание операций, их количество не превышает 200000. 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 выходного файла

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

Пример

rmq2.in	rmq2.out
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 Е. Парковка

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

На кольцевой парковке есть n мест пронумерованных от 1 до n . Есть два вида событий прибытие машину на парковку и отъезд машины с парковки. Если машина приезжает на парковку, а её место занято, то она едет далее по кругу и встаёт на первое свободное место.

Формат входного файла

В первой строке входного файла находится два числа n и m — размер парковки и количество запросов ($1 \leq n, m \leq 100000$). В следующих m строках находятся события. Каждая из этих строк имеет следующий вид:

- `enter x` — приехала машина, которая хочет встать на место x . Для каждой такой команды выведите какое место займёт эта машина.
- `exit x` — уехала машина занимавшая место x . Гарантируется, что на этом месте была машина.

Формат выходного файла

Выведите последовательно результаты выполнения всех операций `enter`.

Пример

parking.in	parking.out
3 5	1
enter 1	2
enter 1	3
exit 1	1
enter 2	
enter 2	