

## Содержание

Задача А. Удалите скобки [0.2 sec, 256 mb]	2
Задача В. Калькулятор [0.5 sec, 256 mb]	3
Задача С. Быстрый поиск в массиве [0.5 sec, 256 mb]	4
Задача D. Компоненты связности [0.2 sec, 256 mb]	5
Задача Е. Расстояние между вершинами [1 sec, 256 mb]	6
Задача F. RMQ [1 sec, 256 mb]	7

---

Задачи нужно сдать на языке **C++**. Это единственный доступный вам в системе язык.

Обратите внимание, входные данные лежат в **стандартном потоке ввода** (он же stdin), вывести ответ нужно в **стандартный поток вывода** (он же stdout).

В некоторых задачах большой ввод и вывод.

Вы можете воспользоваться **быстрым вводом-выводом**, **пример**.

**Задача А. Удалите скобки [0.2 sec, 256 mb]**

Дана строка, составленная из круглых скобок. Определите, какое наименьшее количество символов необходимо удалить из этой строки, чтобы оставшиеся символы образовывали правильную скобочную последовательность.

**Формат входных данных**

Во входном файле записана строка из круглых скобок. Длина строки не превосходит 100 000 символов.

**Формат выходных данных**

Выведите единственное целое число — ответ на поставленную задачу.

**Примеры**

stdin	stdout
()()	2
))(((	5
((()))	0

### Задача В. Калькулятор [0.5 sec, 256 mb]

Имеется калькулятор, который выполняет следующие операции:

- Умножить число  $X$  на 2.
- Умножить число  $X$  на 3.
- Прибавить к числу  $X$  единицу.

Определите, какое наименьшее количество операций требуется, чтобы получить из числа 1 число  $N$ .

#### Формат входных данных

Во входном файле написано натуральное число  $N$ , не превосходящее  $10^6$ .

#### Формат выходных данных

В первой строке выходного файла выведите минимальное количество операций. Во второй строке выведите числа, последовательно получающиеся при выполнении операций. Первое из них должно быть равно 1, а последнее  $N$ .

#### Примеры

stdin	stdout
1	0 1
5	3 1 3 4 5
962340	17 1 3 9 27 54 55 165 495 1485 4455 8910 17820 17821 53463 160389 160390 481170 962340

**Задача С. Быстрый поиск в массиве [0.5 sec, 256 mb]**

Дан массив из  $N$  целых чисел. Все числа от  $-10^9$  до  $10^9$ .

Нужно уметь отвечать на запросы вида “Сколько чисел имеют значения от  $L$  до  $R$ ?”.

**Формат входных данных**

Число  $N$  ( $1 \leq N \leq 10^5$ ). Далее  $N$  целых чисел.

Затем число запросов  $K$  ( $1 \leq K \leq 10^5$ ).

Далее  $K$  пар чисел  $L, R$  ( $-10^9 \leq L \leq R \leq 10^9$ ) — собственно запросы.

**Формат выходных данных**

Выведите  $K$  чисел — ответы на запросы.

**Пример**

stdin	stdout
5	5 2 2 0
10 1 10 3 4	
4	
1 10	
2 9	
3 4	
2 2	

**Задача D. Компоненты связности [0.2 сек, 256 mb]**

Вам задан неориентированный граф с  $N$  вершинами и  $M$  ребрами ( $1 \leq N \leq 20\,000$ ,  $1 \leq M \leq 200\,000$ ). В графе отсутствуют петли и кратные ребра.

Определите компоненты связности заданного графа.

**Формат входных данных**

Граф задан во входном файле следующим образом: первая строка содержит числа  $N$  и  $M$ . Каждая из следующих  $M$  строк содержит описание ребра — два целых числа из диапазона от 1 до  $N$  — номера концов ребра.

**Формат выходных данных**

На первой строке выходного файла выведите число  $L$  — количество компонент связности заданного графа. На следующей строке выведите  $N$  чисел из диапазона от 1 до  $L$  — номера компонент связности, которым принадлежат соответствующие вершины. Компоненты связности следует занумеровать от 1 до  $L$  произвольным образом.

**Пример**

stdin	stdout
4 2	2
1 2	1 1 2 2
3 4	

### Задача E. Расстояние между вершинами [1 сек, 256 mb]

Дан взвешенный неориентированный граф. Требуется найти вес минимального пути между двумя вершинами.

#### Формат входных данных

Первая строка входного файла содержит два натуральных числа  $n$  и  $m$  — количество вершин и ребер графа соответственно. Вторая строка входного файла содержит натуральные числа  $s$  и  $t$  — номера вершин, длину пути между которыми требуется найти ( $1 \leq s, t \leq n$ ,  $s \neq t$ ).

Следующие  $m$  строк содержат описание ребер по одному на строке. Ребро номер  $i$  описывается тремя натуральными числами  $b_i$ ,  $e_i$  и  $w_i$  — номера концов ребра и его вес соответственно ( $1 \leq b_i, e_i \leq n$ ,  $0 \leq w_i \leq 100$ ).

$$n \leq 100\,000, m \leq 200\,000.$$

#### Формат выходных данных

Первая строка выходного файла должна содержать одно натуральное число — вес минимального пути между вершинами  $s$  и  $t$ .

Если путь из  $s$  в  $t$  не существует, выведите -1.

#### Пример

stdin	stdout
4 4 1 3 1 2 1 3 4 5 3 2 2 4 1 4	3

### Задача F. RMQ [1 сек, 256 mb]

Дан массив  $a[1..n]$ . Требуется написать программу, обрабатывающую два типа запросов.

- “max  $l r$ ”. Найти максимум в массиве  $a$  от  $l$ -ой ячейки до  $r$ -ой включительно.
- “add  $l r v$ ”. Прибавить значение  $v$  к каждой ячейке  $a[i]$  от  $l$ -ой до  $r$ -ой включительно.

#### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $q$  ( $1 \leq n, q \leq 10^5$ ) – длина массива и число запросов соответственно. Вторая строка содержит  $n$  целых чисел  $a_1, \dots, a_n$  ( $|a_i| \leq 10^5$ ), задающих соответствующие значения массива. Следующие  $q$  строк содержат запросы.

В зависимости от типа запрос может иметь вид либо “max  $l r$ ”, либо “add  $l r v$ ”.

$$1 \leq l \leq r \leq n, |v| \leq 10^5.$$

#### Формат выходных данных

Для каждого запроса вида “max  $l r$ ” требуется в отдельной строке выдать значение соответствующего максимума.

#### Примеры

stdin	stdout
5 3	3
1 2 3 4 -5	7
max 1 3	
add 1 2 5	
max 1 3	