

Задача А. RMQ

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

Реализуйте структуру данных, которая на данном массиве из N целых чисел позволяет узнать максимальное значение на этом массиве и индекс элемента, на котором достигается это максимальное значение.

Формат входных данных

В первой строке вводится натуральное число N ($1 \leq N \leq 10^5$) – количество элементов в массиве. В следующей строке содержатся N целых чисел, не превосходящих по модулю 10^9 – элементы массива. Гарантируется, что в массиве нет одинаковых элементов. Далее идет число K ($0 \leq K \leq 10^5$) – количество запросов к структуре данных. Каждая из следующих K строк содержит два целых числа l и r ($1 \leq l \leq r \leq N$) – левую и правую границы отрезка в массиве для данного запроса.

Формат выходных данных

Для каждого из запросов выведите два числа: наибольшее значение среди элементов массива на отрезке от l до r и индекс одного из элементов массива, принадлежащий отрезку от l до r , на котором достигается этот максимум.

Пример

стандартный ввод	стандартный вывод
5	7 1
7 3 1 6 4	6 4
3	1 3
1 5	
2 4	
3 3	

Задача В. Поиск максимума

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

Реализуйте структуру данных для эффективного вычисления номера максимального из нескольких подряд идущих элементов массива.

Формат входных данных

В первой строке вводится одно натуральное число N ($1 \leq N \leq 100\,000$) — количество чисел в массиве.

Во второй строке вводятся N чисел от 1 до 100 000 — элементы массива.

В третьей строке вводится одно натуральное число K ($1 \leq K \leq 3\,000\,000$) — количество запросов на вычисление максимума.

В следующих K строках вводится по два числа — номера левого и правого элементов отрезка массива (считается, что элементы массива нумеруются с единицы).

Формат выходных данных

Для каждого запроса выведите индекс максимального элемента на указанном отрезке массива. Если максимальных элементов несколько, выведите любой их них.

Числа выводите в одну строку через пробел.

Пример

стандартный ввод	стандартный вывод
5	3
2 2 2 1 5	5
2	
2 3	
2 5	

Задача С. Нолики

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

Дедус любит давать своим ученикам сложные задачки. На этот раз он придумал такую задачу: Рейтинг всех его учеников записан в массив A . Запросы Дедуса таковы:

1. Изменить рейтинг i -го ученика на число x
2. Найти максимальную последовательность подряд идущих ноликов в массиве A на отрезке $[l, r]$.

Помогите бедным фиксикам избежать зверского наказания за нерешение задачи на этот раз.

Формат входных данных

В первой строке входного файла записано число N ($1 \leq N \leq 500\,000$) – количество учеников. Во второй строке записано N чисел – их рейтинги, числа по модулю не превосходящие 1000 (по количеству задач, которые ученик решил или не решил за время обучения). В третьей строке записано число M ($1 \leq M \leq 50\,000$) – количество запросов. Каждая из следующих M строк содержит описание запросов:

«UPDATE i x » – обновить i -ый элемент массива значением x ($1 \leq i \leq N$, $|x| \leq 1000$)

«QUERY l r » – найти длину максимальной последовательности из нулей на отрезке с l по r . ($1 \leq l \leq r \leq N$)

Формат выходных данных

В выходной файл выведите ответы на запросы «QUERY» в том же порядке, что и во входном файле

Пример

стандартный ввод	стандартный вывод
5	2
328 0 0 0 0	1
5	1
QUERY 1 3	
UPDATE 2 832	
QUERY 3 3	
QUERY 2 3	
UPDATE 2 0	

Задача D. Катый ноль

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

Реализуйте эффективную структуру данных, позволяющую изменять элементы массива и вычислять индекс k -го слева нуля на данном отрезке в массиве.

Формат входных данных

В первой строке вводится одно натуральное число N ($1 \leq N \leq 200\,000$) — количество чисел в массиве. Во второй строке вводятся N чисел от 0 до 100 000 — элементы массива. В третьей строке вводится одно натуральное число M ($1 \leq M \leq 200\,000$) — количество запросов. Каждая из следующих M строк представляет собой описание запроса. Сначала вводится одна буква, кодирующая вид запроса (s — вычислить индекс k -го нуля, u — обновить значение элемента). Следом за s вводится три числа — левый и правый концы отрезка и число k ($1 \leq k \leq N$). Следом за u вводятся два числа — номер элемента и его новое значение.

Формат выходных данных

Для каждого запроса s выведите результат. Все числа выводите в одну строку через пробел. Если нужного числа нулей на запрашиваемом отрезке нет, выводите -1 для данного запроса.

Пример

стандартный ввод	стандартный вывод
5	4
0 0 3 0 2	
3	
u 1 5	
u 1 0	
s 1 5 3	

Замечание

TL для Python 8 секунд

Задача Е. Число возрастающих подпоследовательностей

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

Задана последовательность из n чисел a_1, a_2, \dots, a_n . Подпоследовательностью длины k этой последовательности называется набор индексов i_1, i_2, \dots, i_k , удовлетворяющий неравенствам $1 \leq i_1 < i_2 < \dots < i_k \leq n$. Подпоследовательность называется возрастающей, если выполняются неравенства $a_{i_1} < a_{i_2} < \dots < a_{i_k}$.

Необходимо найти число возрастающих подпоследовательностей наибольшей длины заданной последовательности a_1, \dots, a_n . Так как это число может быть достаточно большим, необходимо найти остаток от его деления на $10^9 + 7$.

Формат входных данных

Первая строка входного файла содержит целое число n ($1 \leq n \leq 10^5$). Вторая строка входного файла содержит n целых чисел: a_1, a_2, \dots, a_n . Все a_i не превосходят 10^9 по абсолютной величине.

Формат выходных данных

В выходной файл выведите ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
5 1 2 3 4 5	1
6 1 1 2 2 3 3	8

Задача F. Противник слаб

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 5 секунд
Ограничение по памяти: 256 мегабайт

Римляне снова наступают. На этот раз их гораздо больше чем персов, но Шапур готов победить их. Он говорит: «Лев никогда не испугается сотни овец».

Не смотря на это, Шапур должен найти слабость римской армии чтобы победить ее. Как вы помните, Шапур — математик, поэтому он определяет насколько слаба армии как число — степень слабости.

Шапур считает, что степень слабости армии равна количеству таких троек i, j, k , что $i < j < k$ и $a_i > a_j > a_k$, где a_x — сила человека, стоящего в строю на месте с номером x .

Помогите Шапуру узнать, насколько слаба армия римлян.

Формат входных данных

В первой строке записано одно целое число n ($3 \leq n \leq 10^6$) — количество солдат в римской армии. Следующая строка содержит n целых чисел a_i ($1 \leq i \leq n, 1 \leq a_i \leq 10^9$) — силы людей в римской армии.

Формат выходных данных

Выведите одно число — степень слабости римской армии.

Примеры

стандартный ввод	стандартный вывод
3 3 2 1	1
3 2 3 1	0
4 10 8 3 1	4
4 1 5 4 3	1

Задача G. Команды

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	256 мегабайт

Т-школа собирает команду на Волшебную командную олимпиаду школьников по программированию (ВКОШП). В команде должно быть k участников, всего же в Т-школе n учеников. У каждого ученика Т-школы есть уникальный показатель крутости p_i — число от 1 до n . На награждении ВКОШПа участников одной команды перечисляют в алфавитном порядке (для простоты будем считать, что имя школьника — это его порядковый номер, то есть участники заданы в алфавитном порядке). Назовем хорошей командой такую, в которой участников на награждении будут перечислять в порядке убывания их крутости. Посчитайте, сколько различных хороших команд из k участников можно получить. Так как это число может быть очень большим, вам необходимо лишь найти его по модулю $10^9 + 7$.

Формат входных данных

Первая строка входного файла содержит два целых числа n и k ($1 \leq n \leq 10^5$, $2 \leq k \leq 10$).

Следующая строка содержит n чисел p_1, p_2, \dots, p_n ($1 \leq p_i \leq n$). Все p_i попарно различны.

Формат выходных данных

Единственная строка выходного файла должна содержать одно число, равное количеству хороших команд по модулю $10^9 + 7$.

Пример

стандартный ввод	стандартный вывод
4 3	4
4 3 2 1	

Задача Н. Дерево отрезков с операцией на отрезке

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.5 секунд
Ограничение по памяти: 256 мегабайт

Реализуйте эффективную структуру данных для хранения элементов и увеличения нескольких подряд идущих элементов на одно и то же число.

Формат входных данных

В первой строке вводится одно натуральное число N ($1 \leq N \leq 100\,000$) — количество чисел в массиве.

Во второй строке вводятся N чисел от 0 до 100 000 — элементы массива.

В третьей строке вводится одно натуральное число M ($1 \leq M \leq 30\,000$) — количество запросов.

Каждая из следующих M строк представляет собой описание запроса. Сначала вводится одна буква, кодирующая вид запроса (g — получить текущее значение элемента по его номеру, a — увеличить все элементы на отрезке).

Следом за g вводится одно число — номер элемента.

Следом за a вводится три числа — левый и правый концы отрезка и число add , на которое нужно увеличить все элементы данного отрезка массива ($0 \leq add \leq 100\,000$).

Формат выходных данных

Выведите в одну строку через пробел ответы на каждый запрос g .

Пример

стандартный ввод	стандартный вывод
5	4
2 4 3 5 2	2
5	14
g 2	5
g 5	
a 1 3 10	
g 2	
g 4	

Задача I. Ближайшее большее число справа

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

Дан массив a из n чисел. Нужно обрабатывать запросы:

0. $\text{set}(i, x)$ – присвоить новое значение элементу массива $a[i] = x$;
1. $\text{get}(i, x)$ – найти $\min k: k \geq i$ и $a_k \geq x$.

Формат входных данных

Первая строка входных данных содержит два числа: длину массива n и количество запросов m ($1 \leq n, m \leq 200\,000$).

Во второй строке записаны n целых чисел – элементы массива a ($0 \leq a_i \leq 200\,000$).

Следующие m строк содержат запросы, каждый запрос содержит три числа t, i, x . Первое число t равно 0 или 1 – тип запроса. $t = 0$ означает запрос типа **set**, $t = 1$ соответствует запросу типа **get**, $1 \leq i \leq n$, $0 \leq x \leq 200\,000$. Элементы массива нумеруются с единицы.

Формат выходных данных

На каждый запрос типа **get** на отдельной строке выведите соответствующее значение k . Если такого k не существует, выведите -1 .

Пример

стандартный ввод	стандартный вывод
4 5	1
1 2 3 4	3
1 1 1	-1
1 1 3	2
1 1 5	
0 2 3	
1 1 3	

Задача J. Сережа и скобочки

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

У Сережи есть строка s длины n , состоящая из символов «(» и «)».

Сереже нужно ответить на m запросов, каждый из которых характеризуется двумя целыми числами l_i, r_i . Ответом на i -ый запрос является длина наибольшей правильной скобочной подпоследовательности последовательности $s_{l_i}, s_{l_i+1}, \dots, s_{r_i}$. Помогите Сереже ответить на все запросы.

Формат входных данных

Первая строка содержит последовательность символов без пробелов s_1, s_2, \dots, s_n ($1 \leq n \leq 10^6$). Каждый символ это либо «(», либо «)». Вторая строка содержит целое число m ($1 \leq m \leq 10^5$) количество запросов. Каждая из следующих m строк содержит пару целых чисел. В i -ой строке записаны числа l_i, r_i , ($1 \leq l_i \leq r_i \leq n$) — описание i -го запроса.

Формат выходных данных

Выведите ответ на каждый запрос в отдельной строке. Ответы выводите в порядке следования запросов во входных данных.

Пример

стандартный ввод	стандартный вывод
()()()()	0
7	0
1 1	2
2 3	10
1 2	4
1 12	6
8 12	6
5 11	
2 10	

Замечание

Подпоследовательностью длины $|x|$ строки $s = s_1 s_2 \dots s_{|s|}$ (где $|s|$ — длина строки s) называется строка $x = s_{k_1} s_{k_2} \dots s_{k_{|x|}}$ ($1 \leq k_1 < k_2 < \dots < k_{|x|} \leq |s|$).

Правильной скобочной последовательностью называется скобочная последовательность, которую можно преобразовать в корректное арифметическое выражение путем вставок между ее символами символов «1» и «+». Например, скобочные последовательности «()()», «((()))» — правильные (полученные выражения: «(1)+(1)», «((1+1)+1)»), а «()» и «(» — нет.

Для третьего запроса искомая последовательность будет «()».

Для четвертого запроса искомая последовательность будет «()()()».

Задача К. Студенты должны страдать

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

За великую службу великому ученому Намрегу великое государство Яинамрег решило выдать великий титул великого Арега. По историческим причинам, каждый, кого посвящают в Ареги, обязан подготовить для студентов задачу. Намрег к данной традиции подошел с энтузиазмом — он решил подготовить задачу, решение для которой сложно и придумать, и написать. Ее условие придумать он смог, а вот сил рук (все-таки ученые думают головой, а не руками) для написания авторского решения ему не хватило. Будучи вашим давним другом, он обратился к вам за помощью. Напишите решение к следующей задаче:

Дана строка s из n строчных английских букв. Поступают запросы двух типов:

- вместо p -го символа строки записать символ c ;
- сказать период подстроки с l -го по r -й символы строки s .

Формат входных данных

В первой строке дано число n ($5 \leq n \leq 10^5$) — размер строки s . Во второй строке дана сама строка s . В третьей строке дано число q ($5 \leq q \leq 10^5$) — количество запросов. В следующих q строках даны описания запросов:

- запросы изменения символа имеют вид $1 p c$ ($1 \leq p \leq n$);
- запросы поиска периода подстроки имеют вид $2 l r$ ($1 \leq l \leq r \leq n$).

Формат выходных данных

Выведите ответы на все запросы поиска периода.

Пример

стандартный ввод	стандартный вывод
10	10
abacabaaba	3
11	4
2 1 10	1
2 5 10	2
2 6 9	5
2 7 8	2
1 4 b	3
2 1 4	1
2 1 5	
2 1 6	
2 5 10	
1 2 a	
2 1 3	