

Задача А. Хип ли?

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Структуру данных Неар можно реализовать на основе массива.

Для этого должно выполняться *основное свойство Неар'a*, которое заключается в следующем.
Для каждого $1 \leq i \leq n$ выполняются следующие условия:

- Если $2i \leq n$, то $a[i] \leq a[2i]$
- Если $2i + 1 \leq n$, то $a[i] \leq a[2i + 1]$

Дан массив целых чисел. Определите является ли он Неар'ом.

Формат входных данных

Первая строка входного файла содержит целое число n ($1 \leq n \leq 10^5$). Вторая строка содержит n целых чисел по модулю не превосходящих $2 \cdot 10^9$.

Формат выходных данных

Выведите «YES», если массив является Неар'ом и «NO» в противном случае.

Примеры

| stdin | stdout |
|----------------|--------|
| 5 1 0 1 2 0 | NO |
| 5 1 3 2 5 4 | YES |

Задача В. Разрезанные таблицы

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

Дан массив из n чисел. Требуется написать программу, которая будет отвечать на запросы следующего вида: найти минимум на отрезке между u и v включительно.

Формат входных данных

В первой строке входного файла даны три натуральных числа n, m ($1 \leq n \leq 10^5, 1 \leq m \leq 10^7$) и a_1 ($0 \leq a_1 < 16\,714\,589$) — количество элементов в массиве, количество запросов и первый элемент массива соответственно. Вторая строка содержит два натуральных числа u_1 и v_1 ($1 \leq u_1, v_1 \leq n$) — первый запрос.

Элементы a_2, a_3, \dots, a_n задаются следующей формулой:

$$a_{i+1} = (23 \cdot a_i + 21563) \bmod 16714589.$$

Например, при $n = 10, a_1 = 12345$ получается следующий массив: $a = (12345, 305498, 7048017, 11694653, 1565158, 2591019, 9471233, 570265, 13137658, 1325095)$.

Запросы генерируются следующим образом:

$$\begin{aligned} u_{i+1} &= ((17 \cdot u_i + 751 + ans_i + 2i) \bmod n) + 1, \\ v_{i+1} &= ((13 \cdot v_i + 593 + ans_i + 5i) \bmod n) + 1, \end{aligned}$$

где ans_i — ответ на запрос номер i .

Обратите внимание, что u_i может быть больше, чем v_i .

Формат выходных данных

В выходной файл выведите u_m, v_m и ans_m (последний запрос и ответ на него).

Примеры

| <code>sparse.in</code> | <code>sparse.out</code> |
|------------------------|-------------------------|
| 10 8 12345 3 9 | 5 3 1565158 |

Замечание

Пояснение к тесту из примера: запросы и результаты.

| a_1 | a_2 | a_3 | a_4 | a_5 | a_6 | a_7 | a_8 | a_9 | a_{10} |
|-------|--------|---------|----------|---------|---------|---------|--------|----------|----------|
| 12345 | 305498 | 7048017 | 11694653 | 1565158 | 2591019 | 9471233 | 570265 | 13137658 | 1325095 |

| # | u | v | ans |
|---|-----|-----|---------|
| 1 | 3 | 9 | 570265 |
| 2 | 10 | 1 | 12345 |
| 3 | 1 | 2 | 12345 |
| 4 | 10 | 10 | 1325095 |
| 5 | 5 | 9 | 570265 |
| 6 | 2 | 1 | 12345 |
| 7 | 3 | 2 | 305498 |
| 8 | 5 | 3 | 1565158 |

Задача С. Электрички

Имя входного файла: `trains.in`
Имя выходного файла: `trains.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

На вокзале есть K тупиков, куда прибывают электрички. Этот вокзал является их конечной станцией, поэтому электрички, прибыв, некоторое время стоят на вокзале, а потом отправляются в новый рейс (в ту сторону, откуда прибыли).

Дано расписание движения, в котором указаны события прибытия и отбытия для каждой из электричек в хронологическом порядке. Поскольку вокзал — конечная станция, то электричка может стоять на нем довольно долго, в частности, электричка, которая прибывает раньше другой, отправляться обратно может значительно позднее.

Тупики пронумерованы числами от 1 до K . Когда электричка прибывает, ее ставят в свободный тупик с минимальным номером.

Напишите программу, которая по данному расписанию для каждой электрички определит номер тупика, куда прибудет эта электричка.

Формат входных данных

В первой строке вводится число K — количество тупиков ($1 \leq K \leq 20000$). Далее следуют строки, описывающие события прибытия/отбытия электричек. Каждая электричка задаётся своей противоположной конечной станцией — строкой длины не более 15 из латинских букв и знаков подчёркивания. Событие `+city` означает, что прибывает электричка из города `city`, событие `-city` — что эта электричка отправляется обратно. Общее количество электричек, фигурирующих в условии — не более 100000, для каждой фигурирующей электрички присутствуют оба события.

Считается, что в нулевой момент времени все тупики на вокзале свободны.

Формат выходных данных

Выведите по одному числу на каждую электричку — номер тупика, куда её поставят по прибытии. Если тупиков не достаточно для того, чтобы организовать движение электричек согласно расписанию, выведите число 0 и город первой из электричек, которая не сможет прибыть на вокзал.

Примеры

| <code>trains.in</code> | <code>trains.out</code> |
|---|--|
| 3 +bologoe +moscow -bologoe +stpetersburg -stpetersburg +samara +saratov -moscow -samara -saratov | bologoe 1 moscow 2 stpetersburg 1 samara 1 saratov 3 |
| 2 +kostroma +sudislavl +newvasyuki -sudislavl -kostroma -newvasyuki | 0 newvasyuki |

Замечание

Использовать только *priorityqueue*, *set* - банится

Задача D. Дорешивание

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Как известно, после обеда в ЛКШ проходит много интересных мероприятий, но все равно каждый ЛКШОнок старается хотя бы ненадолго заглянуть в комповник, чтобы дорешать задачи, не сделанные во время практики.

В этом году погода стоит особо жаркая, поэтому в комповнике очень душно и важно следить за тем, чтобы в комповнике не находилось одновременно очень много школьников. Поэтому завуч записал время прихода и ухода из комповника каждого ЛКШОнка.

Теперь завуч хочет узнать, сколько ЛКШАт встретил в комповнике каждый ЛКШОнок.

Формат входных данных

В первой строке записано количество ЛКШАт N ($1 \leq N \leq 10^5$). В i -й из следующих N строк через пробел записаны целые числа S_i и T_i ($0 \leq S_i \leq T_i \leq 10^9$) — время прихода в комповник и ухода из него i -го ЛКШОнка.

Формат выходных данных

Программа должна вывести N целых чисел, i -е число должно быть равно количеству ЛКШАт, которых встретил в комповнике i -й ЛКШОнок.

Если в некоторый момент времени один ЛКШОнок приходит в комповник, а другой уходит из него, то они встречаются друг с другом.

Примеры

| stdin | stdout |
|-------|--------|
| 4 | 3 |
| 1 10 | 3 |
| 2 5 | 2 |
| 5 6 | 2 |
| 1 4 | |

Замечание

Ограничение времени работы для языка Python — 4 секунды.

Задача Е. Интеллектуальный отпуск

Имя входного файла: e.in
Имя выходного файла: e.out
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

Туристическая отрасль в этом сезоне столкнулась с серьёзными сложностями. Добросовестные туроператоры ищут новые рекламные ходы для продажи своих туров. Как известно, наиболее благоприятная для отдыха погода меняется плавно, причём не только от одного дня к другому, но и в течение суток.

Для большинства туристических направлений есть многолетние посекундные результаты измерений различных климатических параметров, например, температуры или влажности. У каждого человека своё понимание того, насколько различными могут быть подобные значения во время отпуска, но всех интересуют непрерывные туры как можно большей продолжительности.

Пусть мы зафиксировали туристическое направление и некоторый климатический параметр. Будем называть изменчивостью тура разницу между максимальным и минимальным значением выбранного параметра за всё время поездки. Для каждого туриста известно максимальное приемлемое значение изменчивости k_i .

Даны результаты измерений некоторого климатического параметра на одном из курортов и значения k_i для нескольких туристов. Требуется для каждого из них определить максимальный диапазон, подходящий для отпуска.

Формат входных данных

В первой строке входного файла находится целое число N ($1 \leq N \leq 600\,000$) — количество сделанных измерений. Во второй строке — N целых чисел, по модулю не превосходящих 10^9 — данные посекундных измерений.

В третьей строке входного файла находится число M ($1 \leq M \leq 100$) — количество туристов, для которых необходимо найти оптимальный диапазон. В четвёртой строке — M целых чисел k_1, k_2, \dots, k_M ($0 \leq k_i \leq 10^9$) — максимальная возможная разница между выбранным климатическим параметром в непрерывном диапазоне дней для каждого из туристов.

Формат выходных данных

В выходной файл для каждого из M запросов в отдельной строке выведите два числа: номер первого измерения диапазона и номер последнего измерения, входящего в диапазон. Нумерация измерений ведётся с единицы. Если для некоторого туриста существует несколько подходящих диапазонов максимальной длины, выведите границы любого из них.

Примеры

| e.in | e.out |
|--------------------|-------|
| 7 | 3 5 |
| 10 1 10 12 11 1 11 | 4 5 |
| 2 | |
| 2 1 | |
| 9 | 3 4 |
| 1 5 2 3 6 4 7 8 9 | 1 9 |
| 6 | 7 9 |
| 1 10 2 4 5 0 | 2 6 |
| | 1 6 |
| | 1 1 |

Задача F. Друзья и последовательности

Имя входного файла: `friends.in`
Имя выходного файла: `friends.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Майк и !Майк соперничают еще со школьных лет, они противоположны во всем что делают, кроме программирования. Сегодня у них возникла проблема, которую сами друзья сами решить не могут, но вместе с вами — кто знает?

Каждый из них знает две последовательности n чисел a и b . По запросу в виде пары целых чисел (l, r) Майк может сразу сообщить значение $\max_{i=l}^r a_i$, а !Майк — значение $\min_{i=l}^r b_i$.

Предположим, что робот задает им каждый из возможных различных запросов в виде пары целых чисел (l, r) ($1 \leq l \leq r \leq n$) (то есть он сделает ровно $n(n+1)/2$ запросов) и считает, сколько раз их ответы на один и тот же запрос совпадают, то есть для скольких пар выполняется $\max_{i=l}^r a_i = \min_{i=l}^r b_i$.

Сколько случаев совпадения посчитает робот?

Формат входных данных

В первой строке содержится единственное целое число n ($1 \leq n \leq 200\,000$).

Во второй строке содержатся n целых чисел a_1, a_2, \dots, a_n ($-10^9 \leq a_i \leq 10^9$) — элементы последовательности a .

В третьей строке содержатся n целых чисел b_1, b_2, \dots, b_n ($-10^9 \leq b_i \leq 10^9$) — элементы последовательности b .

Формат выходных данных

Выведите одно целое число — количество совпадений ответов, которые посчитает робот, то есть для скольких пар выполняется $\max_{i=l}^r a_i = \min_{i=l}^r b_i$.

Примеры

| friends.in | friends.out |
|---------------------------------|-------------|
| 6 1 2 3 2 1 4 6 7 1 2 3 2 | 2 |
| 3 3 3 3 1 1 1 | 0 |