

Задача А. Скобки

Имя входного файла: `brackets.in`
Имя выходного файла: `brackets.out`
Ограничение по времени: 0.5 second
Ограничение по памяти: 64 megabytes

Требуется определить, является ли правильной данная последовательность круглых, квадратных и фигурных скобок.

Формат входных данных

В единственной строке входного файла записано подряд N скобок ($1 \leq N \leq 10^5$).

Формат выходных данных

В выходной файл вывести «YES», если данная последовательность является правильной, и «NO» в противном случае.

Примеры

<code>brackets.in</code>	<code>brackets.out</code>
<code>()</code>	YES
<code>([])</code>	YES

Замечание

Скобочная последовательность называется правильной, если ее можно получить из какого-либо математического выражения вычеркиванием всех символов, кроме скобок.

Формальное определение правильной скобочной последовательности таково:

1. Пустая последовательность является правильной.
2. Если A — правильная скобочная последовательность, то (A) , $[A]$ и $\{A\}$ — правильные скобочные последовательности.
3. Если A и B — правильные скобочные последовательности, то AB — правильная скобочная последовательность.

Задача В. Минимум на стеке

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

Вам требуется реализовать структуру данных, выполняющую следующие операции:

1. Добавить элемент x в конец структуры.
2. Удалить последний элемент из структуры.
3. Выдать минимальный элемент в структуре.

Формат входных данных

В первой строке входного файла задано одно целое число n — количество операций ($1 \leq n \leq 10^6$). В следующих n строках заданы сами операции. В i -ой строке число t_i — тип операции (1, если операция добавления, 2, если операция удаления, 3, если операция минимума). Если задана операция добавления, то через пробел записано целое число x — элемент, который следует добавить в структуру ($-10^9 \leq x \leq 10^9$). Гарантируется, что перед каждой операцией удаления или нахождения минимума структура не пуста.

Формат выходных данных

Для каждой операции нахождения минимума выведите одно число — минимальный элемент в структуре. Ответы разделяйте переводом строки.

Примеры

<code>stack.in</code>	<code>stack.out</code>
8	-3
1 2	2
1 3	2
1 -3	
3	
2	
3	
2	
3	

Задача С. Наибольший общий делитель

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

Наибольшим общим делителем непустого набора натуральных чисел A называется максимальное натуральное число d , такое что оно является одновременно делителем всех чисел множества A .

Задан массив натуральных чисел $[a_1, a_2, \dots, a_n]$ и число k . Требуется выбрать в нем подмассив из k подряд идущих элементов $[a_l, a_{l+1}, \dots, a_{l+k-1}]$, чтобы их наибольший общий делитель был как можно больше, и вывести этот наибольший общий делитель.

Формат входных данных

Первая строка ввода содержит два целых числа n и k ($2 \leq n \leq 500\,000$, $2 \leq k \leq n$).

Вторая строка содержит n натуральных чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^{18}$).

Формат выходных данных

Выведите одно натуральное число — максимальное возможное значение наибольшего общего делителя элементов подмассива длины k заданного массива.

Примеры

стандартный ввод	стандартный вывод
10 4 2 3 4 8 12 6 12 18 4 3	6
2 2 12 18	6

Задача D. Гоблины и шаманы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	64 мегабайта

Гоблины Мглистых гор очень любят ходить к своим шаманам. Так как гоблинов много, к шаманам часто образуются очень длинные очереди. А поскольку много гоблинов в одном месте быстро образуют шумную толку, которая мешает шаманам проводить сложные медицинские манипуляции, последние решили установить некоторые правила касательно порядка в очереди.

Обычные гоблины при посещении шаманов должны вставать в конец очереди. Привилегированные же гоблины, знающие особый пароль, встают ровно в ее середину, причем при нечетной длине очереди они встают сразу за центром.

Так как гоблины также широко известны своим непочтительным отношением ко всяческим правилам и законам, шаманы попросили вас написать программу, которая бы отслеживала порядок гоблинов в очереди.

Формат входных данных

В первой строке входных данных записано число N ($1 \leq N \leq 2 \cdot 10^5$) - количество запросов к программе. Следующие N строк содержат описание запросов в формате:

- «+ i » — гоблин с номером i ($1 \leq i \leq N$) встает в конец очереди.
- «* i » — привилегированный гоблин с номером i встает в середину очереди.
- «-» — первый гоблин из очереди уходит к шаманам. Гарантируется, что на момент такого запроса очередь не пуста.

Формат выходных данных

Для каждого запроса типа «-» программа должна вывести номер гоблина, который должен зайти к шаманам.

Примеры

стандартный ввод	стандартный вывод
7	1
+ 1	2
+ 2	3
-	
+ 3	
+ 4	
-	
-	

Задача Е. Дюбели и сверла

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

Петя хочет повесить картину на стену. Для этого ему нужно просверлить в стене дырку, вбить в нее дюбель и вкрутить в него саморез. Петя покопался в кладовке и нашел n сверел и m дюбелей. Петя хочет найти сверло и дюбель одного радиуса. Однако, таких может не быть, в этом случае он хочет подобрать сверло и дюбель так, чтобы разность их диаметров была как можно меньше. Помогите Пете.

Формат входных данных

В первой строке входного файла заданы целые числа n и m ($1 \leq n, m \leq 10^5$). Во второй строке заданы n целых чисел — диаметры сверел. В следующей строке заданы m целых чисел — диаметры дюбелей. Диаметры заданы в неубывающем порядке, все диаметры — числа от 1 до 10^9 .

Формат выходных данных

Выведите минимальную возможную разницу диаметров сверла и дюбеля.

Примеры

<code>drill.in</code>	<code>drill.out</code>
3 2 1 8 15 5 6	2
3 3 1 3 5 3 4 6	0

Задача F. Город Че

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

В центре города Че есть пешеходная улица — одно из самых популярных мест для прогулок жителей города. По этой улице очень приятно гулять, ведь вдоль улицы расположено n забавных памятников.

Девочке Маше из города Че нравятся два мальчика из ее школы, и она никак не может сделать выбор между ними. Чтобы принять окончательное решение, она решила назначить обоим мальчикам свидание в одно и то же время. Маша хочет выбрать два памятника на пешеходной улице, около которых мальчики будут ее ждать. При этом она хочет выбрать такие памятники, чтобы мальчики не увидели друг друга. Маша знает, что из-за тумана мальчики увидят друг друга только в том случае, если они будут на расстоянии не более r метров.

Маше заинтересовалась, а сколько способов есть выбрать два различных памятника для организации свиданий.

Формат входных данных

В первой строке входного файла находятся два целых числа n и r ($2 \leq n \leq 300\,000$, $1 \leq r \leq 10^9$) — количество памятников и максимальное расстояние, на котором мальчики могут увидеть друг друга.

Во второй строке задано n положительных чисел d_1, \dots, d_n , где d_i — расстояние от i -го памятника до начала улицы. Все памятники находятся на разном расстоянии от начала улицы. Памятники приведены в порядке возрастания расстояния от начала улицы ($1 \leq d_1 < d_2 < \dots < d_n \leq 10^9$).

Формат выходных данных

Выведите одно число — число способов выбрать два памятника для организации свиданий.

Примеры

<code>che.in</code>	<code>che.out</code>
4 4 1 3 5 8	2

Замечание

В приведенном примере Маша может выбрать памятники 1 и 4 или памятники 2 и 4.

Задача G. Отрезок с максимальной суммой

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

Дан массив целых чисел. Найти отрезок этого массива с максимальной суммой.

Формат входных данных

В первой строке дано натуральное число n ($1 \leq n \leq 100\,000$) — размер массива. Во второй строке через пробел перечислены элементы массива. Числа по модулю не превышают 10 000.

Формат выходных данных

Выведите три числа — индекс начала отрезка, индекс конца и саму максимальную сумму. Отрезок не может быть пустым, но отрезок из одного элемента не считается пустым. Массив индексируется с единицы. Если ответов несколько — выведите любой.

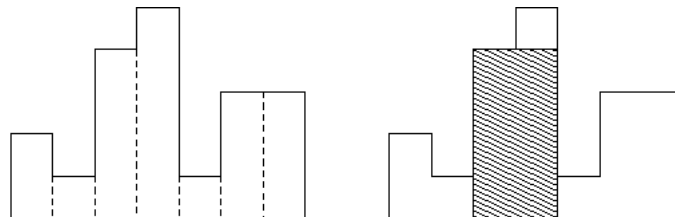
Примеры

<code>max-segment.in</code>	<code>max-segment.out</code>
5 -1 2 3 -2 5	2 5 8

Задача Н. Гистограмма

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

Гистограмма является многоугольником, сформированным из последовательности прямоугольников, выровненных на общей базовой линии. Прямоугольники имеют равную ширину, но могут иметь различные высоты. Например, фигура на рисунке показывает гистограмму, которая состоит из прямоугольников с высотами 2, 1, 4, 5, 1, 3, 3. Все прямоугольники на этом рисунке имеют ширину, равную 1.



Обычно гистограммы используются для представления дискретных распределений, например, частоты символов в текстах. Отметьте, что порядок прямоугольников очень важен. Вычислите область самого большого прямоугольника в гистограмме, который также находится на общей базовой линии. На рисунке справа заштрихованная фигура является самым большим выровненным прямоугольником на изображенной гистограмме.

Формат входных данных

В первой строке входного файла записано число N ($0 \leq N \leq 10^6$) — количество прямоугольников гистограммы. Затем следует N целых чисел h_1, \dots, h_n , где $0 \leq h_i \leq 10^9$. Эти числа обозначают высоты прямоугольников гистограммы слева направо. Ширина каждого прямоугольника равна 1.

Формат выходных данных

Выведите площадь самого большого прямоугольника в гистограмме. Помните, что этот прямоугольник должен быть на общей базовой линии.

Примеры

стандартный ввод	стандартный вывод
7 2 1 4 5 1 3 3	8
4 1000 1000 1000 1000	4000

Задача I. 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 J. A + B

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

Это интерактивная задача. Ваше решение должно строго следовать описанному ниже протоколу взаимодействия с интерактором.

«Вот бы все задачи в контекстах были такие же простые, как “A + B”...»...

Выведите сумму чисел a и b .

Числа a и b загаданы интерактором и вам не сообщаются. Чтобы получить какую-то информацию про числа a и b , вы можете сообщить интерактору любое целое неотрицательное число x , после чего интерактор вернет вам число, равное

$$(a \& b \& x) + (a | b | x) + (a \oplus b \oplus x)$$

Здесь за $\&$, $|$ и \oplus обозначены операции побитового «и» (**and**), «или» (**or**) и «исключающего или» (**xor**) соответственно. Таким образом, на каждый ваш запрос интерактор сообщает вам сумму трех величин, получаемых из a , b и вашего числа x с помощью каждой из данных операций.

Сделав не более 5 запросов к интерактору, найдите и выведите значение величины $a + b$.

Формат входных данных

Интерактор загадывает два целых неотрицательных числа a и b ($0 \leq a, b \leq 10^9$). На ввод вашей программе ничего не поступает.

Формат выходных данных

Ваша программа может посылать интерактору запросы в формате «? x » ($0 \leq x \leq 10^{12}$). В ответ на каждый запрос такого вида интерактор сообщает вашей программе число $(a \& b \& x) + (a | b | x) + (a \oplus b \oplus x)$.

Если переданный в запросе x не укладывается в указанные ограничения или если превышает ограничение в 5 запросов на один тест, интерактор выводит «-1» (без кавычек) и завершается, а ваше решение получает вердикт WA (Wrong Answer). Считав «-1», ваша программа должна завершиться, иначе может быть получен некорректный вердикт TL (Time Limit Exceeded).

Если ваша программа готова дать ответ на задачу, следует вывести запрос формата «! x », где x — предполагаемое значение суммы $a + b$. Считав такой запрос, интерактор выставляет решению вердикт WA или OK в зависимости от правильности ответа и завершается. Соответственно, после вывода такого запроса ваша программа тоже должна завершаться.

После каждого запроса необходимо сбрасывать буфер вывода (`cout.flush()` в C++, `sys.stdout.flush()` в Python) и выводить перевод строки. После каждого ответа на запрос интерактор также выводит перевод строки. Лимит в 5 запросов не учитывает запрос второго типа, то есть можно задать 5 запросов формата «? x » и 1 запрос формата «! x ».

Примеры

стандартный ввод	стандартный вывод
3	? 1 ! 2

Замечание

В примере из условия делается запрос для $x = 1$, на что интерактор сообщает, что $(a \& b \& 1) + (a | b | 1) + (a \oplus b \oplus 1) = 3$.

На основе этой информации решение делает предположение, что $a = b = 1$. Простым перебором всех a и b , для которых $a | b | 1 \leq 3$ можно доказать, что другие значения a и b не могут дать ту же сумму, поэтому решение сразу выдает $1 + 1 = 2$ в качестве ответа.