

Задача А. Проверка ПСП

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дана строка, состоящая из круглых, квадратных и фигурных скобок. Нужно проверить, является ли она правильной скобочной последовательностью.

Формат входных данных

Во входном файле записана скобочная последовательность длиной не более 10 000 символов.

Формат выходных данных

Выведите **YES**, если скобочная последовательность является правильной, и **NO** в противном случае.

Примеры

<code>stdin</code>	<code>stdout</code>
<code>([])</code>	YES
<code>([]</code>	NO

Задача В. Постфиксная запись

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 1 second
Ограничение по памяти: 64 megabytes

В постфиксной записи (или обратной польской записи) операция записывается после двух операндов. Например, сумма двух чисел A и B записывается как $A B +$. Запись $B C + D *$ обозначает привычное нам $(B + C) * D$, а запись $A B C + D * +$ означает $A + (B + C) * D$. Достоинство постфиксной записи в том, что она не требует скобок и дополнительных соглашений о приоритете операторов для своего чтения.

Дано выражение в обратной польской записи. Определите его значение.

Формат входных данных

В единственной строке записано выражение в постфиксной записи, содержащее однозначные числа и операции $+$, $-$, $*$. Строка содержит не более 100 чисел и операций.

Формат выходных данных

Необходимо вывести значение записанного выражения. Гарантируется, что результат выражения, а также результаты всех промежуточных вычислений по модулю меньше 2^{31} .

Примеры

stdin	stdout
8 9 + 1 7 - *	-102

Задача С. Кофейные колечки

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

Саша пьёт кофе всё время из одной и той же кружки и не моет её в течение дня. Когда кофе находится в кружке, то оставляет кофейное кольцо. Когда в кружку подливается кофе, то все кольца ниже уровня нового кофе растворяются.

В течение дня Саша может выполнять следующие действия:

- наливать в кружку X миллилитров кофе;
- выливать из кружки X миллилитров кофе.

Известно, что утром Саша взяла чистую кружку, а в течение дня n раз доливала или выливала кофе. Посчитайте, сколько кофейных колец будет на кружке Саши в конце дня.

Кольцо на уровне 0, полученное после выливания всего кофе, тоже учитывается.

Гарантируется, что в любой момент времени в кружке находится не меньше 0 и не больше 10^9 миллилитров кофе.

Формат входных данных

В первой строке дано одно число n ($0 < n < 500\,000$) — количество действий Саши.

Далее следуют n строк вида $-X$ или $+X$, описывающие процесс выливания из кружки или доливания в кружку X миллилитров кофе соответственно.

Все значения X — целые и не равны 0.

Формат выходных данных

Выведите одно число — количество кофейных колец на кружке Саши в конце дня.

Пример

стандартный ввод	стандартный вывод
3 +5 -3 -1	3

Замечание

В тестовом примере кофейные колечки останутся на отметках 5 мл, 2 мл и 1 мл.

Задача D. Великое Лайнландское переселение

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Лайнландия представляет из себя одномерный мир, являющийся прямой, на котором располагаются N городов, последовательно пронумерованных от 0 до $N - 1$. Направление в сторону от первого города к нулевому названо западным, а в обратную — восточным.

Когда в Лайнландии неожиданно начался кризис, все были жители мира стали испытывать глубокое смятение. По всей Лайнландии стали ходить слухи, что на востоке живётся лучше, чем на западе.

Так и началось Великое Лайнландское переселение. Обитатели мира целыми городами отправились на восток, покинув родные улицы, и двигались до тех пор, пока не приходили в город, в котором средняя цена проживания была меньше, чем в родном.

Формат входных данных

В первой строке дано число N ($2 \leq N \leq 10^5$) - количество городов в Лайнландии. Во второй строке дано N чисел a_i ($0 \leq a_i \leq 10^9$) - средняя цена проживания в городах с нулевого по $(N - 1)$ -ый соответственно.

Формат выходных данных

Для каждого города в порядке с нулевого по $(N - 1)$ -ый выведите номер города, в который переселятся его изначальные жители. Если жители города не остановятся в каком-либо другом городе, отправившись в Восточное Бесконечное Ничто, выведите -1.

Примеры

стандартный ввод	стандартный вывод
10 1 2 3 2 1 4 2 5 3 1	-1 4 3 4 -1 6 9 8 9 -1

Задача Е. Минимум в окне

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

Дан ряд из N чисел. Требуется вывести минимумы из каждых K последовательных чисел.

Формат входных данных

В первой строке вводится одно натуральное число N , не превосходящее 100000. Во второй строке вводится одно натуральное число K , не превосходящее N . В следующих N строках вводится по одному натуральному числу, не превосходящему 100000.

Формат выходных данных

Выведите $N - K + 1$ чисел: минимум из первых K чисел, минимум из следующих K чисел (начиная со второго) и т.д.

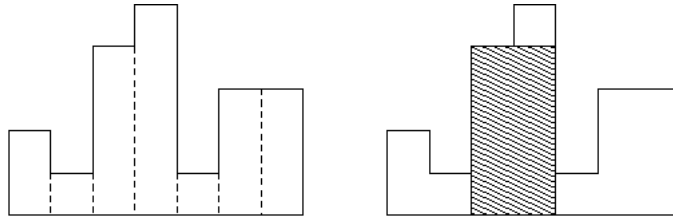
Примеры

window.in	window.out
5	3
3	2
5	1
3	
5	
2	
1	

Задача F. 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 G. Большой, белый, очень прямоугольный

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

В прямоугольной таблице клетки раскрашены в белый и черный цвета. Найти в ней прямоугольную область белого цвета, состоящую из наибольшего количества ячеек.

Формат входных данных

Во входном файле записана сначала высота N , а затем ширина M таблицы ($1 \leq N \leq 5000$), ($1 \leq M \leq 5000$), а затем записано N строк по M чисел в каждой строке, где 0 означает, что соответствующая клетка таблицы выкрашена в белый цвет, а 1 – что в черный.

Формат выходных данных

В выходной файл вывести одно число — количество клеток, содержащихся в наибольшем по площади белом прямоугольнике.

Примеры

white.in	white.out
5 6 1 0 0 0 1 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0	9
4 4 0 0 0 0 0 1 0 1 0 0 0 0 1 1 0 0	4

Задача Н. Гоблины и шаманы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2.5 секунд
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» — гoblin с номером i ($1 \leq i \leq N$) встает в конец очереди.
- «* i» — привилегированный гoblin с номером i встает в середину очереди.
- «-» — первый гoblin 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 I. Марсианская парикмахерская

Имя входного файла: `saloon.in`
 Имя выходного файла: `saloon.out`
 Ограничение по времени: 1 second
 Ограничение по памяти: 64 megabytes

Пока я не поспал, «сегодня» не наступило

мистер Грин

В парикмахерской работает один мастер. Он тратит на одного клиента ровно 20 минут, а затем сразу переходит к следующему, если в очереди кто-то есть, либо ожидает, когда придет следующий клиент.

Даны времена прихода клиентов в парикмахерскую (в том порядке, в котором они приходили).

Также у каждого клиента есть характеристика, называемая *степенью нетерпения*. Она показывает, сколько человек может максимально находиться в очереди перед клиентом, чтобы он дождался своей очереди и не ушел раньше. Если в момент прихода клиента в очереди находится больше людей, чем степень его нетерпения, то он решает не ждать своей очереди и уходит. Клиент, который обслуживается в данный момент, также считается находящимся в очереди.

Требуется для каждого клиента указать время его выхода из парикмахерской.

Формат входных данных

В первой строке вводится натуральное число N , не превышающее 10^5 — количество клиентов.

В следующих N строках вводятся времена прихода клиентов — по два числа, обозначающие часы и минуты (часы — от 0 до 16 000 000, минуты — от 0 до 59) и степень его нетерпения (неотрицательное целое число не большее 10^5) — максимальное количество человек, которое он готов ждать впереди себя в очереди. Времена указаны в порядке возрастания (все времена различны).

Если для каких-то клиентов время окончания обслуживания одного клиента и время прихода другого совпадают, то можно считать, что в начале заканчивается обслуживание первого клиента, а потом приходит второй клиент.

Формат выходных данных

В выходной файл выведите N пар чисел: времена выхода из парикмахерской 1-го, 2-го, ..., N -го клиента (часы и минуты). Если на момент прихода клиента человек в очереди больше, чем степень его нетерпения, то нужно считать, что время его ухода равно времени прихода.

Примеры

saloon.in	saloon.out
3	10 20
10 0 0	10 40
10 1 1	10 2
10 2 1	
5	1 20
1 0 100	2 20
2 0 0	2 1
2 1 0	2 40
2 2 3	2 3
2 3 0	

Задача J. Атака титанов

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2.5 секунд
Ограничение по памяти:	256 мегабайт

В этой задаче мы хотим увидеть решение *ЛИНЕЙНОЙ* сложности (т.е. мы не хотим что-то похожее на $O(n \log n)$)

Как известно из манги и аниме «Атака Титанов», выжившие за стенами люди ведут войну с огромными существами - Титанами.

В 843 году отряд разведки обнаружил отряд титанов, движущийся к внешней стене. Для оперативной ликвидации угрозы к защитникам стены (гарнизону) послали гонца с описанием роста каждого из титанов. Благодаря brave инженерам людей были созданы пушки, уничтожающие титанов с одного выстрела, но, к сожалению, наука далека от совершенства, поэтому у пушек есть 2 проблемы:

1. Одна пушка может уничтожать только титанов определённого роста, который заранее неизвестен
2. Идеально откалибровать пушку невозможно, поэтому она устанавливается на определённую высоту и может подниматься не больше чем на **коэффициент калибровки**.

Для примера, если **коэффициент калибровки** равен 5, и пушка установлена на высоту 15, она сможет уничтожать только титанов какой-то конкретной высоты от 15 до 20 включительно (например, 17).

У гарнизона существуют пушки для каждой возможной высоты, но, к сожалению, из-за срочности они могут подвезти к стене только одну из них. Благодаря знанию структур данных защитники стены смогли выбрать пушку, которая сможет **гарантированно** уничтожить как можно больше титанов, и пережили атаку. А сможете ли Вы?

Кроме того, чем выше пушка, тем больше работы предстоит гарнизону, поэтому сообщайте самую низкую пушку из тех, что дают лучший результат.

Формат входных данных

В первой строке подаётся два числа: N - количество титанов ($1 \leq N \leq 10^6$), k - коэффициент калибровки ($1 \leq k \leq 10^6$). Во второй строке находятся N чисел - рост всех каждого из титанов. Рост каждого титана - целое число в диапазоне от 1 до 10^6 включительно

Формат выходных данных

В первой строке выведите одно число: высоту на которую должна быть установлена пушка (минимальную среди лучших)

Во второй строке выведите одно число: количество титанов которое гарантированно будет уничтожено

Примеры

стандартный ввод	стандартный вывод
10 3 1 4 4 3 1 4 4 2 2 3	1 2
3 1 24 18 5	1 0

Задача К. Гробики на колёсиках

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	64 мегабайта

Два преподавателя решили подготовить сложный контекст для своей параллели. Ночь была сложной. И когда они наконец смогли лечь спать, то им приснился кошмар. Они стояли в двух точках числовой прямой, 0 и L ($2 \leq L \leq 10^9$). Им уготована тяжёлая участь — ловить гробики на колёсиках. Помогите им побыстрее избавиться от кошмаров. Между преподавателями находятся N гробиков ($1 \leq N \leq 4 \times 10^5$) на колёсах в различных точках числовой прямой.

У каждого гробика есть три характеристики, описывающие его. Это изначальная координата его положения x_i , направление движения гробика (1 — если он движется в положительном направлении вдоль оси и -1 — если гроб движется в отрицательном направлении вдоль числовой оси). Также у i -го гробика есть собственный вес w_i (вес гробика ограничен $1 \leq w_i \leq 1000$).

Все гробики всегда движутся с постоянной скоростью 1, пока не произойдёт одно из следующих событий:

1. Если i -й гробик достигает преподавателя, то он ловит его и гробик прекращает движение
2. Происходит столкновение, когда два гробика i и j находятся в одной точке, которая не совпадает с точкой, где расположен преподаватель. В этом случае гробику i назначается скорость гробика j и наоборот. Заметим, что гробики могут встретиться в точке, которая не является целым числом.

Пусть T — самое раннее время, когда сумма весов всех гробиков, которые поймали преподаватели, составляет как минимум половину суммы весов всех гробиков. Найдите время T , скажите через сколько они выполнять хотя бы половину работы.

Формат входных данных

Первая строка содержит два числа N и L ($1 \leq N \leq 5 \times 10^4$, $2 \leq L \leq 10^9$). Каждая из последующих N строк содержит три разделённых пробелом целых числа w_i , x_i , d_i . Все x_i различные и удовлетворяют $0 < x_i < L$ и d_i — направление движения ($d_i = -1$ или $d_i = 1$).

Формат выходных данных

Выведите одно целое число — ответ задачи.

Примеры

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