

## 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> | <code>YES</code>    |
| <code>( [ ]</code>     | <code>NO</code>     |

## 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

| <code>stdin</code>         | <code>stdout</code> |
|----------------------------|---------------------|
| <code>8 9 + 1 7 - *</code> | <code>-102</code>   |

## Задача С. 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

| <code>window.in</code> | <code>window.out</code> |
|------------------------|-------------------------|
| 5                      | 3                       |
| 3                      | 2                       |
| 5                      | 1                       |
| 3                      |                         |
| 5                      |                         |
| 2                      |                         |
| 1                      |                         |

## Задача D. 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:

- «+  $i$ » — 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. 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

| <code>saloon.in</code> | <code>saloon.out</code> |
|------------------------|-------------------------|
| 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 F. 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 бравым 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 находится  $N$  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<br>1 4 4 3 1 4 4 2 2 3 | 1<br>2            |
| 3 1<br>24 18 5              | 1<br>0            |

## Задача G. 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 выполнят 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<br>1 1 1<br>2 2 -1<br>3 3 -1 | 3                 |