

## Задача А. Тестирующая система

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

Юный программист Саша написал свою первую тестирующую систему. Он так обрадовался тому, что она скомпилировалась, что решил пригласить школьных друзей на свой собственный контест.

Но в конце тура выяснилось, что система не умеет сортировать команды в таблице результатов. Помогите Саше реализовать эту сортировку.

Команды упорядочиваются по правилам ACM:

- по количеству решённых задач в порядке убывания;
- при равенстве количества решённых задач — по штрафному времени в порядке возрастания;
- при прочих равных — по номеру команды в порядке возрастания.

### Формат входных данных

Первая строка содержит натуральное число  $n$  ( $1 \leq n \leq 100\,000$ ) — количество команд, участвующих в контесте. В  $i$ -й из следующих  $n$  строк записано количество решённых задач  $S$  ( $0 \leq S \leq 100$ ) и штрафное время  $T$  ( $0 \leq T \leq 100\,000$ ) команды с номером  $i$ .

### Формат выходных данных

В выходной файл выведите  $n$  чисел — номера команд в отсортированном порядке.

### Пример

<code>ejudge.in</code>	<code>ejudge.out</code>
5	5 2 1 3 4
3 50	
5 720	
1 7	
0 0	
8 500	

## Задача В. Лишнее число

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

В штаб секретной службы поступило сообщение от одного из агентов. Поступившее сообщение в зашифрованном виде представляет собой последовательность чисел, и лишь специальная программа способна расшифровать его и получить связный текст.

Обычно программа-расшифровщик быстро и бесшумно выдаёт связистам расшифрованный текст, но в этот раз вместо текста от программы поступил сигнал тревоги, свидетельствующий о том, что при пересылке сообщение было взломано или просто повреждено.

Корректное зашифрованное сообщение — это последовательность из  $4 \cdot k$  целых чисел, в котором  $k$  различных чисел присутствуют по 4 раза каждое; для расшифровки даже не важны значения этих чисел, а важен лишь их порядок.

Однако, изучив зашифрованное сообщение, связисты обнаружили, что в нём  $4 \cdot k + 1$  число. При этом ровно одно число является «лишним», то есть при его удалении зашифрованное сообщение становится корректным сообщением из  $4 \cdot k$  чисел (возможно, четыре из них равны удалённому числу).

Связисты решили, что на будущее им нужна программа, которая находит такое «лишнее» число автоматически. Помогите им написать такую программу.

### Формат входных данных

В первой строке входного файла задано число  $N = 4 \cdot k + 1$ , где  $N$  и  $k$  целые, и  $1 \leq k \leq 10\,000$ . В последующих  $N$  строках находятся числа  $A_1, A_2, \dots, A_N$ , по одному числу в каждой — зашифрованное сообщение. Известно, что  $0 \leq A_i \leq 1\,000\,000$ .

### Формат выходных данных

В первую строку выходного файла выведите «лишнее» число из набора  $A_i$ .

### Пример

<code>excess.in</code>	<code>excess.out</code>
5	1
4	
1	
4	
4	
4	

## Задача С. Быстрый поиск в массиве

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

Дан массив из  $n$  целых чисел. Все числа от  $-10^9$  до  $10^9$ .

Нужно уметь отвечать на запросы вида “Сколько чисел имеют значения от  $l$  до  $r$ ?”.

### Формат входных данных

Число  $n$  ( $1 \leq n \leq 10^5$ ). Далее  $n$  целых чисел.

Затем число запросов  $k$  ( $1 \leq k \leq 10^5$ ).

Далее  $k$  пар чисел  $l, r$  ( $-10^9 \leq l \leq r \leq 10^9$ ) — собственно запросы.

### Формат выходных данных

Выведите  $k$  чисел — ответы на запросы.

### Пример

стандартный ввод	стандартный вывод
5 10 1 10 3 4 4 1 10 2 9 3 4 2 2	5 2 2 0

## Задача D. Для любителей статистики

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

Вы никогда не задумывались над тем, сколько человек за год перевозят трамваи города с десятиллионным населением, в котором каждый третий житель пользуется трамваем по два раза в день?

Предположим, что на планете Земля  $n$  городов, в которых есть трамваи. Любители статистики подсчитали для каждого из этих городов, сколько человек перевезено трамваями этого города за последний год. Из этих данных была составлена таблица, в которой города были отсортированы по алфавиту. Позже выяснилось, что для статистики названия городов несущественны, и тогда их просто заменили числами от 1 до  $n$ . Поисковая система, работающая с этими данными, должна уметь быстро отвечать на вопрос, есть ли среди городов с номерами от  $l$  до  $r$  такой, что за год трамваи этого города перевезли ровно  $x$  человек. Вам предстоит реализовать этот модуль системы.

### Формат входных данных

В первой строке дано целое число  $n$ ,  $0 < n < 70\,000$ . В следующей строке приведены статистические данные в виде списка целых чисел через пробел,  $i$ -е число в этом списке — количество человек, перевезенных за год трамваями  $i$ -го города. Все числа в списке положительны и не превосходят  $10^9 - 1$ . В третьей строке дано количество запросов  $q$ ,  $0 < q < 70\,000$ . В следующих  $q$  строках перечислены запросы. Каждый запрос — это тройка целых чисел  $l$ ,  $r$  и  $x$ , записанных через пробел ( $1 \leq l \leq r \leq n$ ,  $0 < x < 10^9$ ).

### Формат выходных данных

Выведите строку длины  $q$ , в которой  $i$ -й символ равен 1, если ответ на  $i$ -й запрос утвердителен, и 0 в противном случае.

### Пример

<code>queries.in</code>	<code>queries.out</code>
5 123 666 314 666 434 5 1 5 314 1 5 578 2 4 666 4 4 713 1 1 123	10101

## Задача Е. Гирлянда

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

Гирлянда состоит из  $n$  лампочек на общем проводе. Один её конец закреплён на заданной высоте  $A$  мм ( $h_1 = A$ ). Благодаря силе тяжести гирлянда прогибается: высота каждой неконцевой лампы на 1 мм меньше, чем средняя высота ближайших соседей ( $h_i = \frac{(h_{i-1} + h_{i+1})}{2} - 1$  для  $1 < i < N$ ). Требуется найти минимальную высоту второго конца  $B$  ( $B = h_n$ ) при условии, что ни одна из лампочек не должна лежать на земле ( $h_i > 0$  для  $1 \leq i \leq N$ ).

### Формат входных данных

В первую строке входного файла содержится два числа  $n$  и  $A$  ( $3 \leq n \leq 1000$ ,  $n$  — целое,  $10 \leq A \leq 1000$ ,  $A$  — вещественное).

### Формат выходных данных

Вывести одно вещественное число  $B$  с двумя знаками после запятой.

### Пример

стандартный ввод	стандартный вывод
8 15	9.75

## Задача F. Количество инверсий

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

Напишите программу, которая для заданного массива  $A = \langle a_1, a_2, \dots, a_n \rangle$  находит количество пар  $(i, j)$  таких, что  $i < j$  и  $a_i > a_j$ .

### Формат входных данных

Первая строка входного файла содержит натуральное число  $n$  ( $1 \leq n \leq 50\,000$ ) — количество элементов массива. Вторая строка содержит  $n$  попарно различных элементов массива  $A$ .

### Формат выходных данных

В выходной файл выведите одно число — ответ на задачу.

### Примеры

<code>inverse.in</code>	<code>inverse.out</code>
4 1 2 4 5	0
4 5 4 2 1	6

## Задача G. K-best

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

У Демьяны есть  $n$  драгоценностей. Каждая из драгоценностей имеет ценность  $v_i$  и вес  $w_i$ . С тех пор, как её мужа Джонни уволили в связи с последним финансовым кризисом, Демьяна решила продать несколько драгоценностей. Для себя она решила оставить лишь  $k$  лучших. Лучших в смысле максимизации достаточно специфического выражения: пусть она оставила для себя драгоценности номер  $i_1, i_2, \dots, i_k$ , тогда максимальной должна быть величина

$$\frac{\sum_{j=1}^k v_{i_j}}{\sum_{j=1}^k w_{i_j}}$$

Помогите Демьяне выбрать  $k$  драгоценностей требуемым образом.

### Формат входных данных

На первой строке  $n$  и  $k$  ( $1 \leq k \leq n \leq 100\,000$ ).

Следующие  $n$  строк содержат пары целых чисел  $v_i, w_i$  ( $0 \leq v_i \leq 10^6, 1 \leq w_i \leq 10^6$ , сумма всех  $v_i$  не превосходит  $10^7$ , сумма всех  $w_i$  также не превосходит  $10^7$ ).

### Формат выходных данных

Выведите  $k$  различных чисел от 1 до  $n$  — номера драгоценностей. Драгоценности нумеруются в том порядке, в котором перечислены во входных данных. Если есть несколько оптимальных ответов, выведите любой.

### Примеры

kbest.in	kbest.out
3 2	1
1 1	2
1 2	
1 3	