

## Содержание

<b>Задачи</b>	<b>2</b>
1 Задача А. Квадратный корень [0.5 секунд, 256 mb]	2
2 Задача В. Корень кубического уравнения [0.5 секунд, 256 mb]	3
3 Задача С. Быстрый поиск в массиве [1 секунда, 256 mb]	4
4 Задача D. Для любителей статистики [1 секунда, 256 mb]	5
5 Задача Е. Вербочки [0.5 секунд, 256 mb]	6
6 Задача F. Одномерный почтальон [2 секунды, 256 mb]	7
7 Задача G. Одномерный финансист [1 секунда, 256 mb]	8
<b>Бонус</b>	<b>9</b>
8 Задача H. Поезда [0.5 секунд, 256 mb]	9
9 Задача I. La cucaracha [0.5 секунд, 256 mb]	10
10 Задача J. Коровы – в стойла! [0.5 секунд, 256 mb]	11

---

В некоторых задачах большой ввод и вывод. Имеет смысл пользоваться супер быстрым вводом-выводом: <http://acm.math.spbu.ru/~sk1/algo/input-output/>

## Задачи

### 1 Задача А. Квадратный корень [0.5 секунд, 256 mb]

Дано целое число  $n$  от 0 до  $2^{64} - 1$ . Ваша задача — найти  $\lfloor \sqrt{n} \rfloor$ .

#### Формат входных данных

Мультитест. На каждой строке по числу  $n$ . Не более 1000 строк.

#### Формат выходных данных

Для каждого  $n$  на отдельной строке ответ на запрос.

#### Примеры

sqrt.in	sqrt.out
0	0
1	1
2	1
3	1
4	2
5	2

## 2 Задача В. Корень кубического уравнения [0.5 секунд, 256 mb]

Дано кубическое уравнение  $ax^3 + bx^2 + cx + d = 0$  ( $a \neq 0$ ). Известно, что у этого уравнения есть ровно один корень. Требуется его найти.

### Формат входных данных

Во входном файле через пробел записаны четыре целых числа:  $-1000 \leq a, b, c, d \leq 1000$ .

### Формат выходных данных

Выведите единственный корень уравнения с точностью не менее 6 знаков после десятичной точки.

### Пример

сubroot.in	сubroot.out
1 -3 3 -1	1
-1 -6 -12 -7	-1.000000

### 3 Задача С. Быстрый поиск в массиве [1 секунда, 256 mb]

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

find3.in	find3.out
5	5 2 2 0
10 1 10 3 4	
4	
1 10	
2 9	
3 4	
2 2	

#### 4 Задача D. Для любителей статистики [1 секунда, 256 mb]

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

queries.in	queries.out
5	10101
123 666 314 666 434	
5	
1 5 314	
1 5 578	
2 4 666	
4 4 713	
1 1 123	

## 5 Задача Е. Вербочки [0.5 секунд, 256 mb]

С утра шел дождь, и ничего не предвещало беды. Но к обеду выглянуло солнце, и в лагерь заглянула СЭС. Пройдя по всем домикам и корпусам, СЭС вынесла следующий вердикт: бельевые веревки в жилых домиках не удовлетворяют нормам СЭС. Как выяснилось, в каждом домике должно быть ровно по одной бельевой веревке, и все веревки должны иметь одинаковую длину. В лагере имеется  $N$  бельевых веревок и  $K$  домиков. Чтобы лагерь не закрыли, требуется так нарезать данные веревки, чтобы среди получившихся веревочек было  $K$  одинаковой длины. Размер штрафа обратно пропорционален длине бельевых веревок, которые будут развешены в домиках. Поэтому начальство лагеря стремится максимизировать длину этих веревочек.

### Формат входных данных

В первой строке заданы два числа —  $N$  ( $1 \leq N \leq 10\,001$ ) и  $K$  ( $1 \leq K \leq 10\,001$ ). Далее в каждой из последующих  $N$  строк записано по одному числу — длине очередной бельевой веревки. Длина веревки задана в сантиметрах. Все длины лежат в интервале от 1 сантиметра до 100 километров включительно.

### Формат выходных данных

В выходной файл следует вывести одно целое число — максимальную длину веревочек, удовлетворяющую условию, в сантиметрах. В случае, если лагерь закроют, выведите 0.

### Пример

ropes.in	ropes.out
4 11 802 743 457 539	200

## 6 Задача F. Одномерный почтальон [2 секунды, 256 mb]

В деревне Печалька живут  $n$  человек, их домики расположены ровно на оси абсцисс. Домик  $i$ -го человека находится в точке  $x_i$ . В деревню приехал и хочет там поселиться почтальон. Координату своего домика  $y$  он хочет выбрать так, чтобы суммарное расстояние от него до всех жителей деревни было минимально возможным. То есть

$$\sum_{i=1}^n |y - x_i| \rightarrow \min$$

Вам дан массив  $x$  из  $n$  случайных целых чисел. Найдите точку  $y$ .

### Формат входных данных

На первой строке число  $n$  ( $1 \leq n \leq 10^7$ ). На второй строке пара целых чисел  $a, b$  от 1 до  $10^9$ , используемая в генераторе случайных чисел.

```
1. unsigned int cur = 0; // беззнаковое 32-битное число
2. unsigned int nextRand24() {
3.     cur = cur * a + b; // вычисляется с переполнениями
4.     return cur >> 8; // число от 0 до 224 - 1.
5. }
6. unsigned int nextRand32() {
7.     unsigned int a = nextRand24(), b = nextRand24();
8.     return (a << 8) ^ b; // число от 0 до 232 - 1.
9. }
```

Элементы массива генерируются последовательно.  $x_i = \text{nextRand32}()$ ;

### Формат выходных данных

Выведите одно число — минимальное суммарное расстояние от точки  $y$  до всех домиков.

### Примеры

postman.in	postman.out
6 239 13	8510257371

### Замечание

Сгенерированный массив: 12, 130926, 3941054950, 2013898548, 197852696, 2753287507.

## 7 Задача G. Одномерный финансист [1 секунда, 256 mb]

В деревне Печалька живут  $n$  человек, их домики расположены ровно на оси абсцисс. Домик  $i$ -го человека находится в точке  $x_i$ . В деревню недавно заселился почтальон. Почтальон построил себе домик в такой точке  $y$ , что суммарное расстояние от него до всех жителей деревни было минимально возможным. А теперь в деревню приехал финансовый аналитик, который привык не только оптимизировать результат, но и оценивать риски. Посмотрев на опыт почтальона, аналитик заметил, что несмотря на то, что сумма минимальна, есть домики очень далеко от дома почтальона. Финансист учел это и свой дом хочет построить в такой точке  $z$ , что

$$\sum_{i=1}^n (z - x_i)^2 \rightarrow \min$$

С почтальоном финансист не дружит, поэтому расстояние до  $y$  в сумме не учитывается. Вам дан массив  $x$  из  $n$  случайных целых чисел. Найдите точку  $z$ .

### Формат входных данных

На первой строке число  $n$  ( $1 \leq n \leq 10^7$ ). На второй строке пара целых чисел  $a, b$  от 1 до  $10^9$ , используемая в генераторе случайных чисел.

```
1. unsigned int cur = 0; // беззнаковое 32-битное число
2. unsigned int nextRand24() {
3.     cur = cur * a + b; // вычисляется с переполнениями
4.     return cur >> 8; // число от 0 до 224 - 1.
5. }
6. unsigned int nextRand32() {
7.     unsigned int a = nextRand24(), b = nextRand24();
8.     return (a << 8) ^ b; // число от 0 до 232 - 1.
9. }
```

Элементы массива генерируются последовательно.  $x_i = \text{nextRand32}()$ ;

### Формат выходных данных

Выведите координату домика финансиста в виде несократимой дроби с положительным знаменателем.

### Примеры

finansist.in	finansist.out
6	3368129374/3
230 10	

### Замечание

Сгенерированный массив: 12, 130926, 3941054950, 2013898548, 197852696, 2753287507.



## Бонус

### 8 Задача Н. Поезда [0.5 секунд, 256 mb]

В связи с участвовавшим числом аварий на железнодорожной ветке Москва–Саратов, руководство железной дороги решило изменить график движения поездов. Тщательный анализ состояния железнодорожного полотна показал, что оптимальным является следующий график движения поездов с учетом остановок на станциях: сначала поезд идет на протяжении  $T_1$  минут со скоростью  $V_1$  метров в минуту, затем  $T_2$  минут со скоростью  $V_2$  метров в минуту, ..., наконец,  $T_N$  минут со скоростью  $V_N$  метров в минуту. В течение некоторых интервалов поезд может стоять (скорость равна 0).

По действующей инструкции обеспечения безопасности движения поездов расстояние между локомотивами двух следующих друг за другом поездов должно быть не менее  $L$  метров. Определите минимально допустимый интервал в минутах между отправлениями поездов, позволяющий им двигаться по этому графику без опасного сближения.

#### Формат входных данных

В первых двух строках входного файла содержится два натуральных числа, задающие минимально допустимое расстояние  $L$  и количество участков пути  $N$  ( $100 \leq L \leq 10\,000$ ,  $1 \leq N \leq 1000$ ). Далее следует  $N$  пар целых чисел  $T_i$  и  $V_i$ , задающих график движения поездов ( $1 \leq T_i \leq 1000$ ,  $0 \leq V_i \leq 1000$ ).

#### Формат выходных данных

В выходной файл необходимо вывести искомый интервал между отправлениями поездов в минутах, не менее чем с тремя верными знаками после десятичной точки.

#### Пример

trains.in	trains.out
1000	27.500
4	
10 0	
30 80	
15 0	
20 100	

## 9 Задача I. La cucaracha [0.5 секунд, 256 mb]

Каждую полночь в квартире ученого Васи начинается ужас. Сотни . . . , о нет! ТЫСЯЧИ тараканов вылазят из каждой дырки к его обеденному столу, уничтожая все крошки и объедки! Вася ненавидит тараканов. Он очень долго думал и сделал Супер-ловушку, которая привлекает всех тараканов в большой зоне после активации. Он планирует активировать ловушку сегодня ночью. Но есть проблема. Эта очень эффективная ловушка с её очень большой зоной работы поглощает огромное количество энергии. Так что, Вася планирует минимизировать время работы этой ловушки. Он собрал информацию о всех местах, в которых живут тараканы. Также он заметил, что все тараканы двигаются только по линиям его скатерти с постоянной скоростью (мы можем предположить, что эта скорость равна 1, так что таракан расположенный в одной из секций, может за 1 единицу времени переместиться на любую соседнюю секцию (по вертикали или горизонтали)). Вася решил активировать его ловушку в одной из секций. Когда ловушка активирована, все тараканы будут двигаться к секции, содержащей ловушку, так быстро, как только смогут. Поэтому в любой момент времени после активации тараканы двигаются к секции, в которой находится ловушка, максимально уменьшая расстояние до неё. Если есть два пути с одинаковым расстоянием, то таракан выберет любой. Напишите программу для Васи, которая выбирает секцию, минимизирующую время, необходимое для уничтожения всех тараканов. Конечно, ваша программа будет считать, что скатерть будет плоскостью с декартовой системой координат и секции — точки с целыми координатами.

### Формат входных данных

В первой строке входного файла содержится число мест, в которых живут тараканы  $N$  ( $1 \leq N \leq 10000$ ). Следующие  $N$  строк содержат  $x$  и  $y$  — координаты мест, в которых живут тараканы (целые числа не больше  $10^9$  по абсолютному значению).

### Формат выходных данных

Вам необходимо вывести только два целых числа  $x$  и  $y$ , не превосходящие по модулю  $10^9$ , — координаты секции, которая минимизирует время работы. Если есть более одно решение — выведите любое из них.

### Пример

cucarach.in	cucarach.out
2	2 2
1 1	
3 3	

## 10 Задача J. Коровы – в стойла! [0.5 секунд, 256 mb]

На прямой расположены стойла, в которые необходимо расставить коров так, чтобы минимальное расстояние между коровами было как можно больше.

### Формат входных данных

В первой строке вводятся числа  $N$  ( $2 < N < 10001$ ) — количество стойл и  $K$  ( $1 < K < N$ ) — количество коров. Во второй строке задаются  $N$  натуральных чисел в порядке возрастания — координаты стойл (координаты не превосходят  $10^9$ ).

### Формат выходных данных

Выведите одно число — наибольшее возможное допустимое расстояние.

### Примеры

cows.in	cows.out
5 3 1 2 3 100 1000	99