

## Содержание

|   |          |
|---|----------|
| <b>Обязательная часть</b>                                   | <b>2</b> |
| 1 Задача А. Количество инверсий [1 секунда, 256 mb]         | 2        |
| 2 Задача В. Квадратный корень [1 секунда, 256 mb]           | 3        |
| 3 Задача С. Корень кубического уравнения [1 секунда, 64 mb] | 4        |
| 4 Задача D. Быстрый поиск в массиве [1 секунда, 64 mb]      | 5        |
| 5 Задача Е. Для любителей статистики [1 секунда, 64 mb]     | 6        |
| 6 Задача F. Одномерный почтальон [1 секунда, 256 mb]        | 7        |
| 7 Задача G. Одномерный финансист [1 секунда, 256 mb]        | 8        |
| <b>Дополнение</b>   | <b>9</b> |
| 8 Задача H. Минимум и максимум [2 секунды, 64 mb]           | 9        |
| 9 Задача I. Том Сойер и его друзья [1 секунда, 64 mb]       | 10       |
| 10 Задача J. K-Best [1 секунда, 256 mb]                     | 11       |

## Обязательная часть

### 1 Задача А. Количество инверсий [1 секунда, 256 mb]

Дан массив случайных целых чисел, нужно найти количество инверсий.

#### Формат входных данных

На первой строке числа  $n$  ( $1 \leq n \leq 1\,000\,000$ ) — размер массива и  $m$  ( $1 \leq m \leq 2^{24}$  числа в массиве от 0 до  $m - 1$ ). 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  $2^{24} - 1$ .
5. }
```

Элементы массива генерируются последовательно.  $x_i = \text{nextRand24}() \% m$ ;

#### Формат выходных данных

Выведите количество инверсий

#### Примеры

| invcnt.in     | invcnt.out |
|---------------|------------|
| 20 5<br>19 18 | 63         |

#### Замечание

Сгенерированный массив: 01142210424031343330.

## 2    Задача В. Квадратный корень [1 секунда, 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        |

### 3    Задача С. Корень кубического уравнения [1 секунда, 64 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

| cubroot.in   | cubroot.out |
|--------------|-------------|
| 1 -3 3 -1    | 1           |
| -1 -6 -12 -7 | -1.000000   |

#### 4    Задача D. Быстрый поиск в массиве [1 секунда, 64 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<br>10 1 10 3 4<br>4<br>1 10<br>2 9<br>3 4<br>2 2 | 5 2 2 0   |

## 5 Задача Е. Для любителей статистики [1 секунда, 64 mb]

Вы никогда не задумывались над тем, сколько человек за год перевозят трамваи города с десятиmillionным 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<br>123 666 314 666 434<br>5<br>1 5 314<br>1 5 578<br>2 4 666<br>4 4 713<br>1 1 123 | 10101       |

## 6 Задача F. Одномерный почтальон [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 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          | 8510257371  |
| 239 13     |             |

### 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  $2^{24} - 1$ .
5. }
6. unsigned int nextRand32() {
7.     unsigned int a = nextRand24(), b = nextRand24();
8.     return (a « 8) ^ b; // число от 0 до  $2^{32} - 1$ .
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<br>230 10  | 3368129374/3  |

### Замечание

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



## Дополнение

### 8 Задача Н. Минимум и максимум [2 секунды, 64 mb]

**Внимание:** в данной задаче запрещается пользоваться уже готовыми структурами данных такими как `set<int>`.

Пусть есть множество целых чисел. Необходимо реализовать структуру данных для их хранения, поддерживающую следующие операции: `GetMin` — извлечение минимума, `GetMax` — извлечение максимума, `Insert(N)` — добавление числа в множество.

#### Формат входных данных

В первой строке входного файла записано одно целое число  $N$  ( $1 \leq N \leq 100\,000$ ) — число запросов к структуре. Затем в  $N$  строках следуют запросы по одному в строке: `GetMin`, `GetMax`, `Insert(A)` — извлечение минимума, максимума и добавление числа  $A$  ( $1 \leq A \leq 2^{31} - 1$ ). Запросы корректны, то есть нет операций извлечения для пустого множества.

#### Формат выходных данных

Для каждого запроса `GetMin` или `GetMax` выведите то число, которое было извлечено.

#### Примеры

| minmax.in   | minmax.out |
|-------------|------------|
| 10          | 1          |
| Insert(100) | 100        |
| Insert(99)  | 1          |
| Insert(1)   | 2          |
| Insert(2)   | 99         |
| GetMin      |            |
| GetMax      |            |
| Insert(1)   |            |
| GetMin      |            |
| GetMin      |            |
| GetMax      |            |

## 9 Задача I. Том Сойер и его друзья [1 секунда, 64 mb]

**Внимание:** в данной задаче запрещается пользоваться деревом отрезком, декартовым деревом и т.д.

Друзья Тома Сойера по очереди красят забор разными красками. Каждый из них красит несколько идущих подряд секций забора в определенный цвет, при этом используемые цвета могут повторяться. Новая краска ложится поверх старой. Для каждой краски вычислите количество секций, которые будут покрашены этой краской после того, как все друзья закончат работу.

### Формат входных данных

В первой строке входного файла содержатся два целых числа:  $N$  ( $1 \leq N \leq 10^9$ ) и  $K$  ( $1 \leq K \leq 50\,000$ ) — количество секций в заборе и количество различных красок соответственно.

Во второй строке содержится единственное число  $M$  ( $0 \leq M \leq 50\,000$ ) — количество друзей Тома Сойера.

Далее следуют  $M$  строк: в  $i$ -ой строке содержится информация о работе друга, который красил забор  $i$ -ым по счету, а именно 3 целых числа  $c_i, l_i, r_i$  ( $1 \leq c_i \leq K, 1 \leq l_i \leq r_i \leq N$ ) — номер краски, которую использовал  $i$ -й друг, номер первой и номер последней покрашенной секции соответственно.

### Формат выходных данных

Выведите в единственную строку выходного файла  $K$  целых чисел:  $i$ -ое число должно быть равно количеству секций, покрашенных  $i$ -й краской.

### Пример

| paint.in                                     | paint.out |
|--|-----------|
| 5 3<br>4<br>1 3 4<br>2 4 5<br>3 2 3<br>1 5 5 | 1 1 2     |
| 5 3<br>3<br>1 1 5<br>2 2 4<br>1 3 3          | 3 2 0     |

## 10    Задача J. K-Best [1 секунда, 256 mb]

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.

### Пример

| kbest.in | kbest.out |
|----------|-----------|
| 3 2      | 1 2       |
| 1 1      |           |
| 1 2      |           |
| 1 3      |           |