

## Содержание

<b>Must have</b>	<b>2</b>
Задача 5A. Количество инверсий [0.5 sec, 256 mb]	2
Задача 5B. Миллиардеры [0.8 sec, 256 mb]	3
<b>Обязательные задачи</b>	<b>5</b>
Задача 5C. Объединение прямоугольников [0.3 sec, 256 mb]	5
Задача 5D. La cucaracha [0.2 sec, 256 mb]	6
Задача 5E. Q-я порядковая статистика [0.8 sec, 256 mb]	7
Задача 5F. Мега-инверсии [0.2 sec, 256 mb]	8
<b>Дополнительные задачи</b>	<b>9</b>
Задача 5G. Сортировка за линейное время [0.4 sec, 256 mb]	9
Задача 5H. Котята за круглым столом [0.8 sec, 256 mb]	10
Задача 5I. Ворчливые коровы [0.2 sec, 256 mb]	11
Задача 5J. Точки в пространстве [0.7 sec, 256 mb]	12

---

Вы не умеете читать/выводить данные, открывать файлы? Воспользуйтесь **примерами**.

В некоторых задачах большой ввод и вывод. Пользуйтесь **быстрым вводом-выводом**.

В некоторых задачах нужен STL, который активно использует динамическую память (set-ы, map-ы) **переопределение стандартного аллокатора** ускорит вашу программу.

Обратите внимание на компилятор GNU C++11 5.1.0 (TDM-GCC-64) inc, который позволяет пользоваться **дополнительной библиотекой**. Под ним можно сдать **вот это**.

## Must have

### Задача 5А. Количество инверсий [0.5 sec, 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

stdin	stdout
20 5 19 18	63

#### Замечание

Сгенерированный массив: 01142210424031343330.

#### Подсказка по решению

Напишите merge sort ;-)

Запрещается пользоваться деревьями Фенвика, отрезков, treap и т.д.

### Задача 5В. Миллиардеры [0.8 сек, 256 mb]

Возможно, вы знаете, что из всех городов мира больше всего миллиардеров живёт в Москве. Но, поскольку работа миллиардера подразумевает частые перемещения по всему свету, в определённые дни какой-то другой город может занимать первую строчку в таком рейтинге. Ваши приятели из ФСБ, ФБР, МІБ и Шин Бет скинули вам списки перемещений всех миллиардеров за последнее время. Ваш работодатель просит посчитать, сколько дней в течение этого периода каждый из городов мира был первым по общей сумме денег миллиардеров, находящихся в нём.

#### Формат входных данных

В первой строке записано число  $n$  — количество миллиардеров ( $1 \leq n \leq 10\,000$ ). Каждая из следующих  $n$  строк содержит данные на определённого человека: его имя, название города, где он находился в первый день данного периода, и размер состояния. В следующей строке записаны два числа:  $m$  — количество дней, о которых есть данные ( $1 \leq m \leq 50\,000$ ),  $k$  — количество зарегистрированных перемещений миллиардеров ( $0 \leq k \leq 50\,000$ ). Следующие  $k$  строк содержат список перемещений в формате: номер дня (от 1 до  $m-1$ ), имя человека, название города назначения. Вы можете считать, что миллиардеры путешествуют не чаще одного раза в день, и что они отбывают поздно вечером и прибывают в город назначения рано утром следующего дня. Список упорядочен по возрастанию номера дня. Все имена и названия городов состоят не более чем из 20 латинских букв, регистр букв имеет значение. Состояния миллиардеров лежат в пределах от 1 до 100 миллиардов.

#### Формат выходных данных

В каждой строке должно содержаться название города и, через пробел, количество дней, в течение которых этот город лидировал по общему состоянию миллиардеров, находящихся в нём. Если таких дней не было, пропустите этот город. Города должны быть отсортированы по алфавиту (используйте обычный порядок символов: ABC...Zabc...z).

#### Примеры

stdin	stdout
5	Anadyr 5
Abramovich London 15000000000	London 14
Deripaska Moscow 10000000000	Moscow 1
Potantin Moscow 5000000000	
Berezovsky London 2500000000	
Khodorkovsky Chita 1000000000	
25 9	
1 Abramovich Anadyr	
5 Potantin Courchevel	
10 Abramovich Moscow	
11 Abramovich London	
11 Deripaska StPetersburg	
15 Potantin Norilsk	
20 Berezovsky Tbilisi	
21 Potantin StPetersburg	
22 Berezovsky London	

#### Замечание

Если упорядочить события по времени, то

London : 1  
Anadyr : 5  
Moscow : 1  
London : 13

что означает, что сперва один день Лондон был на первом месте и так далее...

#### **Подсказка по решению**

Это задача на STL. Не бойтесь им пользоваться!

`unordered_map<string, int>` поможет (кстати, он сам хеширует строки).

## Обязательные задачи

### Задача 5С. Объединение прямоугольников [0.3 сек, 256 mb]

На плоскости задано  $N$  прямоугольников с вершинами в точках с целыми координатами и сторонами, параллельными осям координат. Необходимо найти площадь их объединения.

#### Формат входных данных

В первой строке входного файла указано число  $N$  ( $0 \leq N \leq 1500$ ). В следующих  $N$  строках заданы по 4 целых числа  $x_1, y_1, x_2, y_2$  — сначала координаты левого нижнего угла прямоугольника, потом правого верхнего ( $0 \leq x_1 \leq x_2 \leq 10^9, 0 \leq y_1 \leq y_2 \leq 10^9$ ). Обратите внимание, что прямоугольники могут вырождаться в отрезки и даже в точки.

#### Формат выходных данных

В выходной файл выведите единственное число — ответ на задачу.

#### Пример

stdin	stdout
3 1 1 3 5 5 2 7 4 2 4 6 7	23

### Задача 5D. La cucaracha [0.2 sec, 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

stdin	stdout
2	2 2
1 1	
3 3	

### Задача 5Е. Q-я порядковая статистика [0.8 сек, 256 mb]

Вам дан массив из  $n$  случайных целых чисел. Ваша задача — мысленно отсортировать массив и вывести  $q$ -е число в получившемся порядке.

#### Формат входных данных

На первой строке числа  $n, q$ . ( $1 \leq q \leq n \leq 10^7$ ). 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 ответ на запрос.

#### Примеры

stdin	stdout
6 3	197852696
239 13	

#### Замечание

Сгенерированный массив: 12, 130926, 3941054950, 2013898548, 197852696, 2753287507.  
Внимание: нельзя пользоваться стандартными функциями, например, `nth_element`.

### Задача 5F. Мега-инверсии [0.2 sec, 256 mb]

*Инверсией* в перестановке  $p_1, p_2, \dots, p_N$  называется пара  $(i, j)$  такая, что  $i < j$  и  $p_i > p_j$ . Назовем мега-инверсией в перестановке  $p_1, p_2, \dots, p_N$  тройку  $(i, j, k)$  такую, что  $i < j < k$  и  $p_i > p_j > p_k$ . Придумайте алгоритм для быстрого подсчета количества мега-инверсий в перестановке.

#### Формат входных данных

Первая строка входного файла содержит целое число  $N$  ( $1 \leq N \leq 100\,000$ ). Следующие  $N$  чисел описывают перестановку:  $p_1, p_2, \dots, p_N$  ( $1 \leq p_i \leq N$ ), все  $p_i$  попарно различны. Числа разделяются пробелами и/или переводами строк.

#### Формат выходных данных

Единственная строка выходного файла должна содержать одно число, равное количеству мега-инверсий в перестановке  $p_1, p_2, \dots, p_N$ .

#### Примеры

stdin	stdout
4 4 3 2 1	4

#### Замечание

Разобрана на практике почти полностью ; -)

## Дополнительные задачи

### Задача 5G. Сортировка за линейное время [0.4 сек, 256 mb]

Дан массив случайных целых чисел, нужно отсортировать его.

#### Формат входных данных

На первой строке количество тестов  $t$  ( $1 \leq t \leq 200$ ) и число  $n$  ( $1 \leq n \leq 50\,000$ ) — размер массива в каждом из тестов. 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. }
```

Тесты генерируются последовательно.

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

#### Формат выходных данных

Для каждого теста выведите на отдельной строке  $\sum_{i=1}^n x_i \cdot i$ .

#### Примеры

stdin	stdout
1 6	46062181379
239 13	

#### Замечание

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

В этой задаче очень небольшой запас по времени. Если она не сдается, это нормально.

Предполагается решение через или BucketSort, RadixSort.

### Задача 5Н. Котята за круглым столом [0.8 сек, 256 mb]

Котята собрались за круглым столом, чтобы отпраздновать успешное замяукивание. У них есть блюдечки с молоком, которые нужно расставить вдоль границы стола. Каждый котёнок будет лакать молоко из блюдца, которое ближе всего к нему (расстояние измеряется как длина дуги). Требуется выбрать такие целочисленные позиции для блюдец (длина окружности — целое число, на окружности есть точки с координатами 0, 1, 2 и т.д.), что максимальное расстояние среди всех котят до нужного блюдца будет минимальным.

#### Формат входных данных

В первой строке ввода записаны целые числа  $N$ ,  $K$  и  $L$  — количество котят и блюдец, а также длина окружности стола. В следующей строке записано  $N$  целых чисел от 0 до  $(L - 1)$  в возрастающем порядке — позиции котят.  $1 \leq N, K \leq 500\,000$ ,  $1 \leq L \leq 1\,000\,000\,000$ .

#### Формат выходных данных

В первой строке выведите минимально возможное значение максимального расстояния. Во второй выведите  $K$  целых чисел от 0 до  $(L - 1)$  — позиции блюдец в любом из оптимальных решений.

#### Примеры

stdin	stdout
4 2 1000000	5000
0 67000 68000 77000	0 72000

### Задача 51. Ворчливые коровы [0.2 сек, 256 mb]

У фермера Джона  $N$  коров. Каждый вечеро они выстраиваются в очередь, чтобы их подоили. У каждой коровы есть показатель ворчливости — целое число. Показатели ворчливости всех коров попарно различны.

Ворчливые коровы могут повредить оборудование фермы, поэтому фермер Джон хочет переупорядочить их так, чтобы доить коров в порядке возрастания их ворчливости. Фермер может выбрать двух любых коров (не обязательно стоящих рядом) с ворчливостями  $X$  и  $Y$  и поменять их местами за время  $X + Y$ . Эту операцию можно повторять последовательно сколько угодно раз.

Помогите фермеру Джону вычислить минимальное время, которое потребуется для того, чтобы переставить коров в очереди таким образом, чтобы доить их в порядке возрастания ворчливости.

#### Формат входных данных

В первой строке ввода задано целое число  $N$  — количество коров ( $1 \leq N \leq 10\,000$ ). Следующие  $N$  строк содержат по одному числу каждая. Это ворчливости коров в порядке очереди на дойку. Гарантируется, что ворчливости — попарно различные целые числа от 1 до 100 000.

#### Формат выходных данных

Выведите одно число — минимальное время, за которое можно переупорядочить коров в порядке возрастания ворчливости.

#### Пример

stdin	stdout
3	7
2	
3	
1	

#### Пояснение к примеру

В примере три коровы. Изначально первой стоит корова с ворчливостью 2, второй — с ворчливостью 3, а третьей — с ворчливостью 1.

Фермер Джон может сначала поменять коров с ворчливостями 1 и 3 за время  $1+3 = 4$  и из порядка 2 3 1 получить порядок 2 1 3. Затем фермер может поменять коров с ворчливостями 1 и 2 за время  $1 + 2 = 3$  и получить требуемый порядок 1 2 3.

### Задача 5J. Точки в пространстве [0.7 sec, 256 mb]

В пространстве заданы  $n$  точек. Вас очень интересует одна величина — минимальное из попарных расстояний между точками. Именно её вы и должны найти.

#### Формат входных данных

Первая строка ввода содержит единственное число  $n$  — количество точек ( $2 \leq n \leq 50\,000$ ). Следующие  $n$  строк содержат по три целых числа каждая — координаты точек в пространстве. Гарантируется, что все точки различны. Координаты не превышают  $10^6$  по абсолютной величине.

#### Формат выходных данных

В первой строке выведите единственное вещественное число  $d$  — минимальное расстояние — с точностью не менее 5 знаков. Во второй строке выведите пару целых чисел — номера точек, расстояние между которыми совпадает с ответом. Если таких пар несколько, выведите любую пару.

#### Пример

stdin	stdout
5	1.4142135624
1 1 0	4 3
1 0 1	
0 1 1	
0 0 0	
2 2 2	

#### Замечание

Есть решение за  $\mathcal{O}(n \log n)$ . Решение за  $\mathcal{O}(n \log^2 n)$  также получит ОК.