

## Содержание

<b>01.Base [5/5]</b>	<b>3</b>
1 Задача 1A. Сумма двух [1 секунда, 256 mb]	3
2 Задача 1B. Произведение двух [1 секунда, 256 mb]	4
3 Задача 1C. Коллекционер [1 секунда, 256 mb]	5
4 Задача 1D. Разбиения на слагаемые [cpp: 0.5s, java: 1s, 256 mb]	6
5 Задача 1E. Числа Каталана [1 секунда, 256 mb]	7
<b>01.Advanced [1/2]</b>	<b>8</b>
6 Задача 1F. Маленький холодильник [1 секунда, 256 mb]	8
7 Задача 1G. Умножение матриц [1.5 секунд, 256 mb]	9
<b>01.Hard [0/1]</b>	<b>10</b>
8 Задача 1H. Большой холодильник [2.5 секунд, 256 mb]	10
<b>02.Base [3/3]</b>	<b>12</b>
9 Задача 2A. Заезд в ЛКШ [0.5 секунд, 256 mb]	12
10 Задача 2B. Тестирующая система [cpp: 0.8s, java: 1.5s, 256 mb]	13
11 Задача 2C. Количество инверсий [2 секунды, 256 mb]	14
<b>02.Advanced [1/3]</b>	<b>15</b>
12 Задача 2D. Мега-инверсии [0.5 секунд, 256 mb]	15
13 Задача 2E. Умножение чисел [1 секунда, 256 mb]	16
14 Задача 2F. Ближайшие точки [3 секунды, 256 mb]	17
<b>02.Hard [0/1]</b>	<b>18</b>
15 Задача 2G. Точки в пространстве [2 секунды, 256 mb]	18

**Общие правила:**

Дедлайн на задачи 10 дней с 00:00 в четверг по 23:59 в субботу через неделю.

К каждой главе есть более простые задачи (base), посложнее (advanced), и сложные (hard).

В скобках к каждой главе написано сколько любых задач из этой главы нужно сдать.

**Общая информация по курсу:**

Тестирующая система: <http://contest.yandex.ru/contest/640/>

Сайт курса: <http://compscicenter.ru/courses/algorithms-1/2014-autumn/>

Семинары ведет Сергей Владимирович Копелиович,

контакты: [burunduk30@gmail.com](mailto:burunduk30@gmail.com), [vk.com/burunduk1](https://vk.com/burunduk1)

## 01.Base [5/5]

### 1    Задача 1A. Сумма двух [1 секунда, 256 mb]

#### Формат входных данных

В первой строке входного файла расположены два целых числа  $A$  и  $B$ , не превосходящих 1 000 по модулю.

#### Формат выходных данных

Ваша программа должна выдавать в выходной файл одно число — сумму чисел  $A$  и  $B$ .

#### Примеры

sum.in	sum.out
2 3	5
17 -18	-1

## 2    Задача 1В. Произведение двух [1 секунда, 256 mb]

### Формат входных данных

Входной файл состоит из двух целых чисел  $A$  и  $B$ , не превосходящих по модулю  $10^9$ .

### Формат выходных данных

Программа должна выдавать в выходной файл единственное число — произведение чисел  $A$  и  $B$ .

### Пример

product.in	product.out
2 2	4

### 3 Задача 1С. Коллекционер [1 секунда, 256 mb]

В Байтландии за всю её историю было выпущено 15000 различных почтовых марок. Известный коллекционер почтовых марок планирует собрать полную коллекцию марок Байтландии. Какое-то количество марок (возможно, с дубликатами) у него есть на данный момент). По заданному списку марок, имеющихся в наличии, вычислить, какое минимальное количество марок коллекционер должен докупить, чтобы коллекция стала полной.

#### Формат входных данных

Входной файл состоит из двух строк. В первой строке задано одно целое число  $n$  ( $1 \leq n \leq 3\,000\,000$ ) — количество имеющихся на данный момент у коллекционера экземпляров байтландских марок. Во второй строке заданы  $n$  целых чисел, каждое из которых лежит в интервале  $[1, 15000]$  — номер марки, представленной соответствующим экземпляром.

#### Формат выходных данных

Одно число — минимальное количество марок, которое коллекционер должен докупить, чтобы коллекция стала полной.

#### Примеры

collection.in	collection.out
8 3 6 2 2 4 6 3 7	14995

#### 4    Задача 1D. Разбиения на слагаемые [cpp: 0.5s, java: 1s, 256 mb]

Перечислите все разбиения целого положительного числа  $N$  ( $1 \leq N \leq 40$ ) на целые положительные слагаемые. Разбиения должны обладать следующими свойствами:

1. Слагаемые в разбиениях идут в невозрастающем порядке.
2. Разбиения перечисляются в лексикографическом порядке.

##### Пример

partition.in	partition.out
4	1 1 1 1 2 1 1 2 2 3 1 4

## 5 Задача 1Е. Числа Каталана [1 секунда, 256 mb]

Числа Каталана определяются следующим образом:

1.  $C_0 = 1$
2.  $C_n = \sum_{i=0}^{n-1} C_i C_{n-i-1}$

Ваша задача — посчитать  $C_n \bmod m$ .

### Формат входных данных

На первой строке целые числа  $n$  ( $0 \leq n \leq 1000$ ) и  $m$  ( $1 \leq m \leq 10^9$ ).

### Формат выходных данных

Выведите одно целое число —  $C_n \bmod m$ .

### Примеры

catalan.in	catalan.out
5 1000000000	42

## 01.Advanced [1/2]

### 6    Задача 1F. Маленький холодильник [1 секунда, 256 mb]

Дано целое число  $n$  ( $1 \leq n \leq 10^{12}$ ). Нужно найти натуральные  $a, b, c$ :  $abc = n$  и при этом  $2(ab + bc + ca)$  минимально. Т.е. при фиксированном объеме минимизировать площадь поверхности.

#### Формат входных данных

На первой строке число  $n$  ( $1 \leq n \leq 10^{12}$ ).

#### Формат выходных данных

На первой строке четыре целые числа —  $2(ab + bc + ca)$  и  $a, b, c$ .

#### Примеры

refrator.in	refrator.out
120	148 4 6 5



## 7 Задача 1G. Умножение матриц [1.5 секунд, 256 mb]

Даны две квадратных матрицы из целых неотрицательных чисел и целое число  $m$ . Посчитайте произведение матриц по модулю  $m$ .

### Формат входных данных

На первой строке  $n$  ( $1 \leq n \leq 700$ ),  $m$  ( $1 \leq m \leq 10^9$ ).

Следующие  $n$  строк содержат по  $n$  целых чисел от 0 до  $m - 1$  — матрица  $A$ . Следующие  $n$  строк содержат по  $n$  целых чисел от 0 до  $m - 1$  — матрица  $B$ .

### Формат выходных данных

Выведите  $n$  строк по  $n$  целых чисел от 0 до  $m - 1$  в каждой — матрица  $(A \times B) \bmod m$ .

### Примеры

matmul.in	matmul.out
3 2	1 0 1
1 1 1	1 0 1
1 1 1	1 0 1
1 1 1	
1 0 0	
0 1 0	
0 1 1	

### Замечание

Внимание: это оптимизационная задача. У нас есть решение на c++, оно работает **0.76** секунд. Мы не гарантируем, что ваше решение на другом языке пройдет. Если у вас не сдается эта задача — это нормально =)

## 01.Hard [0/1]

### 8    Задача 1Н. Большой холодильник [2.5 секунд, 256 mb]

Вася хочет купить новый холодильник. Он считает, что холодильник должен быть прямоугольным параллелепипедом с целочисленными длинами ребер. Вася рассчитал, что для повседневного пользования ему понадобится холодильник объема не меньше  $V$ . Кроме того, Вася по натуре минималист, поэтому объем должен быть и не больше  $V$  — к чему занимать лишнее место в квартире? Определившись с объемом холодильника, Вася столкнулся с новой непростой задачей — чтобы холодильник было проще мыть, при фиксированном объеме  $V$  он должен иметь минимальную площадь поверхности.

Объем и площадь поверхности холодильника с ребрами  $a$ ,  $b$ ,  $c$  равны  $V = abc$  и  $S = 2(ab + bc + ca)$ , соответственно.

Помогите Васе по заданному объему  $V$  найти такие целые длины ребер холодильника  $a$ ,  $b$ ,  $c$ , чтобы объем холодильника был равен  $V$  и при этом его площадь поверхности  $S$  была минимальна.

#### Формат входных данных

В первой строке записано единственное целое число  $t$  ( $1 \leq t \leq 500$ ) — количество наборов данных.

Далее следует описание  $t$  наборов данных. Каждый набор состоит из одного целого числа  $V$  ( $2 \leq V \leq 10^{18}$ ), заданного своим разложением на множители следующим образом.

Пусть  $V = p_1^{a_1} p_2^{a_2} \dots p_k^{a_k}$ , где  $p_i$  — различные простые числа, а  $a_i$  — положительные целые степени.

Тогда в первой строке описания набора данных записано единственное положительное целое число  $k$  — количество различных простых делителей  $V$ . В следующих  $k$  строках записаны простые числа  $p_i$  и их степени  $a_i$ , разделенные пробелом. Все  $p_i$  различны, все  $a_i > 0$ .

#### Формат выходных данных

Выведите  $t$  строк, в  $i$ -й строке выведите ответ на  $i$ -й набор данных — четыре целых числа, записанные через пробел: минимальная возможная площадь поверхности  $S$  и соответствующие длины ребер  $a$ ,  $b$ ,  $c$ . Если вариантов длин ребер, дающих минимальную площадь, несколько, разрешается вывести любой из них. Длины ребер холодильника разрешается выводить в любом порядке.

## Примеры

refrigerator.in	refrigerator.out
3	24 2 2 2
1	70 1 1 17
2 3	148 4 6 5
1	
17 1	
3	
3 1	
2 3	
5 1	

## Замечание

В первом наборе данных примера объем холодильника  $V = 2^3 = 8$ , и минимальную площадь поверхности дадут ребра одинаковой длины.

Во втором наборе данных объем  $V = 17$ , и его можно получить из единственного набора ребер целочисленных длин.

## 02.Base [3/3]

### 9    Задача 2A. Заезд в ЛКШ [0.5 секунд, 256 mb]

Знаете ли вы, как непросто организовать заезд в ЛКШ? Например, в 2010 году нужно было заказать автобусы для целых  $n$  ЛКШат, мечтающих добраться в “Орлёнок” из Саратова. Директору смены сообщили, что можно заказать некоторые из  $m$  автобусов. Он узнал вместимость каждого автобуса и сразу понял, какое минимальное количество автобусов ему нужно заказать, чтобы привезти в лагерь всех ЛКШат. А сможете ли вы так же быстро решить эту задачу?

#### Формат входных данных

В первой строке через пробел записаны целые числа  $n$  и  $m$  ( $1 \leq n \leq 10^6$ ;  $1 \leq m \leq 1000$ ). В следующей строке через пробел записаны  $m$  целых чисел в пределах от 1 до 1000 — вместимости автобусов.

#### Формат выходных данных

В первой строке выведите число  $k$  — минимальное количество автобусов, которое придётся заказать директору. В следующей строке выведите через пробел  $k$  целых чисел — номера автобусов, которые нужно заказать. Автобусы пронумерованы от 1 до  $m$  в том порядке, в которых они перечислены во входных данных. Если возможных решений несколько, выведите любое. Если решения нет, в единственной строке выведите «-1».

#### Примеры

arrival.in	arrival.out
345 5 100 130 190 140 150	3 1 3 4
345 3 100 100 100	-1

## 10    Задача 2В. Тестирующая система [cpp: 0.8s, java: 1.5s, 256 mb]

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

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 контексте. В  $i$ -й из следующих  $n$  строк записано количество решенных 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

ejudge.in	ejudge.out
5 3 50 5 720 1 7 0 0 8 500	5 2 1 3 4

## 11 Задача 2С. Количество инверсий [2 секунды, 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 19 18	63

### Замечание

Сгенерированный массив: 01142210424031343330.

## 02.Advanced [1/3]

### 12 Задача 2D. Мега-инверсии [0.5 секунд, 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

mega.in	mega.out
4 4 3 2 1	4

## 13    Задача 2Е. Умножение чисел [1 секунда, 256 mb]

Требуется перемножить два целых неотрицательных числа.

### Формат входных данных

В двух строках даны два целых неотрицательных числа в 10-чной системе счисления. Максимальная длина числа =  $2^{18}$ .

### Формат выходных данных

Выведите в выходной файл произведение.

### Пример

mul.in	mul.out
13	1300
100	



## 14    Задача 2F. Ближайшие точки [3 секунды, 256 mb]

Дано несколько точек на плоскости. Выведите наименьшее расстояние, которое достигается между какими-то двумя из них.

### Формат входных данных

В первой строке задано число  $N$  ( $2 \leq N \leq 200\,000$ ) — количество точек.

Следующие  $N$  строк содержат координаты точек (целые числа от  $-10^9$  до  $10^9$ ).

### Формат выходных данных

Выведите единственное вещественное число — минимальное расстояние между какими-то двумя из этих точек. Ответ будет считаться корректным, если абсолютная погрешность ответа не будет превышать  $10^{-6}$ .

### Примеры

closest.in	closest.out
2 0 0 3 4	5.0
2 7 7 7 7	0.0
4 0 0 5 6 3 4 7 2	2.8284271247461903

## 02.Hard [0/1]

### 15    Задача 2G. Точки в пространстве [2 секунды, 256 mb]

В пространстве заданы  $n$  точек. Вас очень интересует одна величина — минимальное из парных 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

points3d.in	points3d.out
5	1.4142135624
1 1 0	4 3
1 0 1	
0 1 1	
0 0 0	
2 2 2	