

Содержание

01.Знакомство с системой [4/4]	3
Задача 01A. Сумма двух [0.5 sec (1 sec), 256 mb]	3
Задача 01B. Произведение двух [0.5 sec (1 sec), 256 mb]	4
Задача 01C. Сколько работает? [0.5 sec (1 sec), 256 mb]	5
Задача 01D. Перевернутый массив [0.5 sec (1 sec), 256 mb]	6
01.Обязательные задачи [5/5]	7
Задача 01E. Произведение матриц [0.5 sec (1 sec), 256 mb]	7
Задача 01F. Два числа [0.5 sec (1 sec), 256 mb]	8
Задача 01G. Числа Каталана [0.5 sec (1 sec), 256 mb]	9
Задача 01H. Разбиения на слагаемые [0.5 sec (1 sec), 256 mb]	10
Задача 01I. Коллекционер [0.5 sec (1 sec), 256 mb]	11
01.Дополнительные задачи [0/4]	12
Задача 01J. Тестирование мозга [0.5 sec (1 sec), 256 mb]	12
Задача 01K. Умножение матриц [0.5 sec (1 sec), 256 mb]	13
Задача 01L. Маленький холодильник [0.5 sec (1 sec), 256 mb]	14
Задача 01M. Большой холодильник [0.5 sec (1 sec), 256 mb]	15

Общая информация:

Вход в констест: <http://contest.yandex.ru/contest/1457/>

Дедлайн на задачи: 10 дней, до 19-го сентября 23:59.

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.

Сайт курса: <https://compscicenter.ru/courses/algorithms-1/2015-autumn/>

Семинары ведут Сергей Копелиович (burunduk30@gmail.com, vk.com/burunduk1) и Глеб Леонов (gleb.leonov@gmail.com, vk.com/id1509292)

В каждом условии указан таймлимит для C/C++.

Таймлимит для Java примерно в 2-3 раза больше.

Таймлимит для Python примерно в 5 раз больше.

C++:

Быстрый ввод-вывод.

http://acm.math.spbu.ru/~sk1/algo/input-output/cpp_common.html

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

<http://acm.math.spbu.ru/~sk1/algo/memory.cpp.html>

Java:

Быстрый ввод-вывод.

http://acm.math.spbu.ru/~sk1/algo/input-output/java/java_common.html

01.Знакомство с системой [4/4]

Задача 01А. Сумма двух [0.5 sec (1 sec), 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

Задача 01B. Произведение двух [0.5 sec (1 sec), 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

Задача 01С. Сколько работает? [0.5 сек (1 сек), 256 mb]

Посчитайте $\sum_{k=1}^{k^2 \leq n} \lfloor \frac{n}{k^2} \rfloor$.

Формат входных данных

Целое число n ($1 \leq n \leq 10^{16}$).

Формат выходных данных

Выведите сумму.

Примеры

testtime.in	testtime.out
100	153

Замечание

Это простая задача. Сдайте наивное решение и посмотрите, сколько оно работает.

Какова асимптотика времени работы?

Сдайте задачу под разными компиляторами. Посмотрите, есть ли разница.

Задача 01D. Перевернутый массив [0.5 sec (1 sec), 256 mb]

Переверните массив целых чисел от -10^9 до 10^9 .

Формат входных данных

Массив. Формат смотрите в примере. Длина до 10^6 .

Формат выходных данных

Массив.

Пример

io.in	io.out
3 1 2 -3	-3 2 1

Замечание

Эта задача нужна, чтобы вы оценили, сколько времени работает ваш любимый способ ввода/вывода. В течение курса могут появиться задачи с аналогичным объёмом ввода/вывода, но при этом с time limit всего 0.5 секунд.

01.Обязательные задачи [5/5]

Задача 01Е. Произведение матриц [0.5 sec (1 sec), 256 mb]

Даны две матрицы над \mathbb{Z} . Найти их произведение.

Формат входных данных

На первой строке целые числа n, m, k от 1 до 100, задающие размеры матриц.

Следующие n строк задают матрицу A размера $n \times m$.

Следующие m строк задают матрицу B размера $m \times k$.

Элементы матриц – целые числа от 1 до 99.

Формат выходных данных

Выведите n строк, задающие матрицу $A \cdot B$.

Примеры

easymul.in	easymul.out
3 2 4	5388 3332 3290 4168
46 36	12010 8064 7077 9614
95 92	6836 1807 5149 4059
87 1	
78 20 59 46	
50 67 16 57	

Задача 01F. Два числа [0.5 sec (1 sec), 256 mb]

Даны два целых числа A и B ($1 \leq A, B \leq 100$). Найдите два таких целых числа X и Y , что выполнено равенство $AX + BY = 1$.

Формат входных данных

Во входном файле записаны два числа A и B , разделённые пробелом.

Формат выходных данных

В выходной файл выведите два числа X и Y , разделённые пробелом. Требуется, чтобы выполнялись неравенства $|X| \leq 10\,000$, $|Y| \leq 10\,000$. Если правильных ответов несколько, разрешается вывести любой из них. Если же таких чисел не существует, выведите вместо них два нуля.

Примеры

numbers.in	numbers.out
2 3	2 -1
4 6	0 0
100 51	-5075 9951

Задача 01G. Числа Каталана [0.5 sec (1 sec), 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 10000000000	42

Задача 01Н. Разбиения на слагаемые [0.5 сек (1 sec), 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

Задача 01I. Коллекционер [0.5 sec (1 sec), 256 mb]

В Бйтландии за всю её историю было выпущено 15000 различных почтовых марок. Известный коллекционер почтовых марок планирует собрать полную коллекцию марок Бйтландии. 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 бйтландских 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

01.Дополнительные задачи [0/4]

Задача 01J. Тестирование мозга [0.5 sec (1 sec), 256 mb]

Посчитайте $\sum_{k=1}^{k^2 \leq n} \lfloor \frac{n}{k^2} \rfloor$.

Формат входных данных

Целое число n ($1 \leq n \leq 10^{19}$).

Формат выходных данных

Выведите сумму.

Примеры

testbrain.in	testbrain.out
100	153

Замечание

Это не очень простая задача...

Задача 01К. Умножение матриц [0.5 sec (1 sec), 256 mb]

Даны две квадратные 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 =)

Задача 01L. Маленький холодильник [0.5 sec (1 sec), 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

Задача 01M. Большой холодильник [0.5 sec (1 sec), 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.