

Содержание

Разные задачи	2
1 Задача А. Числа Каталана	2
2 Задача В. Разложение числа	3
3 Задача С. Клумба	4
4 Задача D. Бублики	5
5 Задача Е. Перестановки	6
6 Задача F. Удалите скобки	7
7 Задача G. Калькулятор	8
8 Задача H. Обратная перестановка	9

Разные задачи

1 Задача А. Числа Каталана

Имя входного файла: `catalan.in`
Имя выходного файла: `catalan.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

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

<code>catalan.in</code>	<code>catalan.out</code>
5 1000000000	42

2 Задача В. Разложение числа

Имя входного файла: `prime.in`
Имя выходного файла: `prime.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Напишите программу, которая по данному натуральному числу n выводит все его простые натуральные делители с учетом кратности.

Формат входных данных

Программа получает на вход одно целое число n ($1 \leq n < 2^{31}$).

Формат выходных данных

Программа должна вывести все простые натуральные делители числа n с учетом кратности в порядке неубывания.

Примеры

<code>prime.in</code>	<code>prime.out</code>
6	2 3

3 Задача С. Клумба

Имя входного файла: `flowers.in`
Имя выходного файла: `flowers.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

На круглой клумбе перед главным корпусом растет N круглых цветков, которые целиком помещаются на этой клумбе. Цветки имеют разную высоту, поэтому соответствующие круги могут пересекаться. Перед отъездом из ЛКШ школьники решили взять с собой на память горсть земли из клумбы. Администрация пошла навстречу ЛКШатам и согласилась максимально уменьшить радиус клумбы, не меняя положения ее центра, и раздать остаток земли всем желающим. При этом на новой клумбе цветы также должны помещаться целиком.

Формат входных данных

В первой строке входного файла содержатся три целых числа x , y , R — координаты центра клумбы и ее радиус. Во второй строке записано натуральное число N — количество цветков, не превышающее 10^5 . В каждой из следующих N строк записано по три целых числа x_i , y_i , R_i — координаты центра и радиус i -ого цветка. Все координаты и радиусы не превышают 1000 по модулю, все радиусы положительные. Гарантируется, что все цветки лежат внутри клумбы.

Формат выходных данных

В выходной файл выведите с точностью до пяти знаков после запятой единственное вещественное число — новый радиус клумбы.

Примеры

<code>flowers.in</code>	<code>flowers.out</code>
0 0 2 1 0 0 1	1
0 1 10 2 0 0 1 2 3 3	5.82843
0 1 10 2 0 0 1 0 4 7	10

4 Задача D. Бублики

Имя входного файла: `doughnuts.in`
Имя выходного файла: `doughnuts.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 Мебибайт

В сказочной стране пекарей деньги традиционно называются «бубликами». В этой стране в ходу n различных монет номиналом в v_1, v_2, \dots, v_n бубликов. Номиналы таковы, что для любых двух монет достоинство одной из них делится нацело на достоинство другой.

Вас попросили написать программу для автомата, выдающего суммы бубликов монетами. Для удобства пользователей автомат должен выдавать любую сумму при помощи минимально возможного для этой суммы количества монет.

Считая, что количество монет каждого достоинства в автомате не ограничено, выясните, как ему выдать w бубликов минимальным количеством монет.

Формат входных данных

В первой строке входного файла заданы два целых числа n и w через пробел — количество различных достоинств монет и сумма, которую необходимо набрать, соответственно ($1 \leq n \leq 30$, $1 \leq w \leq 10^9$). Во второй строке заданы n различных целых чисел v_1, v_2, \dots, v_n через пробел — достоинства монет ($1 \leq v_i \leq 10^9$). Известно, что для любых двух различных достоинств одно из них делится на другое. Одно из заданных достоинств равно единице.

Формат выходных данных

В первой строке выходного файла выведите одно число k — минимальное количество монет, при помощи которого можно набрать необходимую сумму. Во второй строке перечислите достоинства этих k монет в любом порядке; сумма этих достоинств должна быть равна w . Если оптимальных ответов несколько, можно вывести любой из них. Тесты в этой задаче таковы, что в оптимальном ответе число k не превысит 10 000.

Примеры

<code>doughnuts.in</code>	<code>doughnuts.out</code>
4 90 1 2 10 50	5 10 10 10 10 50
4 3 1 2 10 50	2 2 1

5 Задача E. Перестановки

Имя входного файла: `perm.in`
Имя выходного файла: `perm.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Во входном файле задано число n ($1 \leq n \leq 8$). Выведите в выходной файл в лексикографическом порядке все перестановки чисел от 1 до n .

Пример

<code>perm.in</code>	<code>perm.out</code>
3	1 2 3 1 3 2 2 1 3 2 3 1 3 1 2 3 2 1

6 Задача F. Удалите скобки

Имя входного файла: `erase.in`
Имя выходного файла: `erase.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Дана строка, составленная из круглых скобок. Определите, какое наименьшее количество символов необходимо удалить из этой строки, чтобы оставшиеся символы образовывали правильную скобочную последовательность.

Формат входных данных

Во входном файле записана строка из круглых скобок. Длина строки не превосходит 100 000 символов.

Формат выходных данных

Выведите единственное целое число — ответ на поставленную задачу.

Примеры

<code>erase.in</code>	<code>erase.out</code>
<code>()()</code>	2
<code>)()((</code>	5
<code>((()())</code>	0

7 Задача G. Калькулятор

Имя входного файла: `calcul.in`
Имя выходного файла: `calcul.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Имеется калькулятор, который выполняет следующие операции:

- Умножить число X на 2.
- Умножить число X на 3.
- Прибавить к числу X единицу.

Определите, какое наименьшее количество операций требуется, чтобы получить из числа 1 число N .

Формат входных данных

Во входном файле написано натуральное число N , не превосходящее 10^6 .

Формат выходных данных

В первой строке выходного файла выведите минимальное количество операций. Во второй строке выведите числа, последовательно получающиеся при выполнении операций. Первое из них должно быть равно 1, а последнее N .

Примеры

<code>calcul.in</code>	<code>calcul.out</code>
1	0 1
5	3 1 3 4 5
962340	17 1 3 9 27 54 55 165 495 1485 4455 8910 17820 17821 53463 160389 160390 481170 962340

8 Задача Н. Обратная перестановка

Имя входного файла: `inverse.in`
Имя выходного файла: `inverse.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Обратной перестановкой к перестановке p_1, p_2, \dots, p_n называется перестановка q_1, q_2, \dots, q_n , в которой q_i — это номер места, на котором стоит число i в перестановке p . К примеру, обратной к перестановке 1 4 2 3 является перестановка 1 3 4 2.

По данной перестановке длины n найдите обратную к ней перестановку.

Формат входных данных

В первой строке входного файла задано целое число n ($1 \leq n \leq 10$). Во второй строке заданы n целых чисел p_1, p_2, \dots, p_n через пробел. Гарантируется, что эти числа образуют перестановку чисел $1, 2, \dots, n$.

Формат выходных данных

В первой строке выходного файла выведите n чисел через пробел — перестановку, обратную данной.

Примеры

<code>inverse.in</code>	<code>inverse.out</code>
1 1	1
2 1 2	1 2
4 1 4 2 3	1 3 4 2