

Общая информация по задачам олимпиады

Доступ к результатам проверки решений задач во время тура

В течение тура по каждой задаче можно отправить не более 40 решений и получить информацию о результатах оценивания решения на тестах жюри.

Требования к программам

Во всех задачах размер файла с исходным кодом решения не должен превышать 256 КБ. В каждой задаче входные данные необходимо считывать из стандартного потока ввода, выходные данные необходимо выводить в стандартный поток вывода.

Процесс тестирования

Перед решением задачи ознакомьтесь с системой оценки решения. Обратите внимание, в некоторых задачах очередная подзадача будет тестироваться, только если пройдены все тесты предыдущих подзадач.

Сложность и порядок задач

Задачи муниципального этапа по информатике упорядочены примерно по возрастанию сложности. Полное решение каждой задачи оценивается в 100 баллов.

Ограничения

Задачи	Ограничение по времени	Ограничение по памяти	Получение результатов во время тура
А. Два альбома	1 секунда	256 МБ	Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи.
В. Факториал	1 секунда	256 МБ	Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи.
С. Сортировка мусора	1 секунда	256 МБ	Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи.
Д. Кенгурёнок и Тигра	1 секунда	256 МБ	Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи.
Е. Простые суммы	1 секунда	256 МБ	Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи.

С результатами проверки решений задач, тестами, решениями жюри, а также письменным разбором задач можно ознакомиться после окончания тура на сайте <http://kpfu.ru/math/olimpiady-dlya-shkolnikov-i-studentov/olimpiady-shkolnikov-po-informatike>

Задача А. Два альбома

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

У вас есть два альбома, в одном n почтовых марок, а в другом — m почтовых марок. В каждом альбоме все марки разные, но среди них есть такие, которые встречаются в обоих альбомах. Такие марки вы хотели бы обменять при встрече с другими филателистами.

У каждой альбомной марки есть свой уникальный номер — целое число от 1 до 10^9 , при этом у одинаковых марок одинаковые номера, а у разных марок — разные номера.

Вам необходимо определить количество совпадающих марок в альбомах, а также номера марок, которые есть и в первом, и во втором альбомах.

Формат входных данных

В первой строке записаны целые числа n и m — количества марок в первом и втором альбомах соответственно ($1 \leq n, m \leq 10^5$).

Во второй строке содержатся n различных целых чисел в диапазоне от 1 до 10^9 включительно — уникальные номера марок первого альбома.

В третьей строке содержатся m различных целых в диапазоне от 1 до 10^9 включительно — уникальные номера марок второго альбома.

Формат выходных данных

В первой строке запишите число k — количество совпадающих марок двух альбомов.

Во второй строке запишите в порядке возрастания k целых чисел — уникальные номера этих совпадающих марок.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
1	30	$1 \leq n, m \leq 5$		баллы
2	30	$1 \leq n, m \leq 20\,000$	1	баллы
3	40	$1 \leq n, m \leq 100\,000$	1, 2	баллы

Примеры

стандартный ввод	стандартный вывод
3 3 10 15 20 1 2 3	0
5 6 2 10 5 13 4 6 2 13 3 10 1	3 2 10 13

Задача В. Факториал

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

На доске записана длинная последовательность цифр — десятичная запись числа $n!$, равного произведению всех целых чисел от 1 до n .

Известный школьный хулиган, лентяй и двоечник Виктор Перестукин стёр одну из цифр в записи этого числа, а вместо неё нарисовал символ #. Запись числа $n!$ была очень длинной, и конечно, Перестукин тут же забыл стёртую цифру, и где она располагалась.

Вскоре выяснилось, что число $n!$ входит в условие олимпиадной задачи по информатике, и теперь нужно срочно восстановить его. После воспитательного разговора с директором школы Виктору удалось вспомнить только, что стёртая цифра была отлична от 0.

Ваша задача — составить программу, которая определяет стёртую цифру в записи числа $n!$.

Формат входных данных

В единственной строке записана непустая строка s , состоящая из цифр от 0 до 9 и символа #. Строка s соответствует десятичной записи некоторого числа вида $n!$ ($1 \leq n \leq 10\,000$), в которой один ненулевой символ заменен на символ #. Максимальная длина строки — 35660 (соответствует записи числа 10000!). Первый (слева) символ в строке s отличен от нуля.

Формат выходных данных

Запишите одну ненулевую цифру, которая должна стоять в записи $n!$ вместо символа #. Если решений несколько, выведите любое из них.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

(Обозначение: m — длина строки s .)

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
1	20	$1 \leq m \leq 3$		баллы
2	20	$1 \leq m \leq 20$	1	баллы
3	30	$1 \leq m \leq 5000$	1, 2	баллы
4	30	$1 \leq m \leq 35660$	1, 2, 3	баллы

Примеры

стандартный ввод	стандартный вывод
2#	4
#20	1

Замечание

Во втором примере правильным ответом будет также число 7. (Если вместо символа # записать цифру 7, получится число $720 = 6!$.)

Задача С. Сортировка мусора

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Мусор — одно из главных экологических бедствий современной экологии. Каждый год в мире выбрасываются миллионы тонн предметов, загрязняющих атмосферу и почву. Самый безопасный способ обращения с отходами — отдельный сбор и последующая его переработка.

Во многих странах каждый вид отходов собирается в отдельный контейнер: пластик, бумага, алюминиевые банки и тому подобное. Если же в одном контейнере находятся разные виды отходов, их нужно предварительно рассортировать.

Перед вами n контейнеров, в которых находятся n различных видов отходов. Необходимо рассортировать мусор по этим же контейнерам так, чтобы в каждом из них остался только один вид отходов. За каждую операцию можно перенести только одну единицу какого-либо вида мусора из одного контейнера в другой.

Вам нужно вычислить наименьшее число операций, необходимое для сортировки мусора.

Формат входных данных

В первой строке записано число n — количество контейнеров и видов мусора ($2 \leq n \leq 10$).

В каждой из следующих n строк записаны n целых чисел $a_{i1}, a_{i2}, \dots, a_{in}$, указывающих соответственно количество отходов вида 1, вида 2, и так далее, наконец, вида n , в контейнере с номером i ($1 \leq i \leq n$). Все числа a_{ij} в диапазоне от 1 до $2 \cdot 10^9$ включительно.

Формат выходных данных

Запишите наименьшее количество операций, необходимое для сортировки мусора по контейнерам.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
1	20	$n = 2$		баллы
2	20	$n = 3$	1	баллы
3	20	$n = 4$	1, 2	баллы
4	40	$1 \leq n \leq 10$	1, 2, 3	баллы

Пример

стандартный ввод	стандартный вывод
3 1 2 1 2 1 1 2 2 2	8

Задача D. Кенгурёнок и Тигра

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Маленький кенгурёнок Ру из книги английского писателя Алана Милна «Винни-Пух» сидит в точке A , и, как все кенгуру, умеет только прыгать и совершенно не умеет бегать. Длина каждого его прыжка постоянная и равна d . Каждый вечер кенгурёнок отправляется в гости к своему приятелю Тигре, который живёт в точке B .

Вам необходимо составить программу, которая определяет минимальный по длине маршрут перемещения кенгурёнка из точки A в точку B .

Формат входных данных

В первой строке записаны 4 целых числа x_A, y_A и x_B, y_B — координаты двух различных точек A и B ($-150 \leq x_A, y_A, x_B, y_B \leq 150$).

Во второй строке одно целое число d — длина прыжка кенгурёнка ($1 \leq d \leq 150$).

Формат выходных данных

В первой строке запишите число n — количество прыжков в минимальном по длине маршруте.

В следующих n строках выведите по два действительных числа — координаты (x_i, y_i) , в которых кенгурёнок окажется после i -го прыжка. Ответ считается верным, если абсолютная или относительная погрешность не превосходит 10^{-5} .

Если минимальных по длине маршрутов несколько, выведите любой из них. Если решения не существует, выведите -1 .

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
1	25	$d > AB$		баллы
2	25	$x_A = x_B$	1	баллы
3	25	$y_A = y_B$	1	баллы
4	25		1, 2, 3	баллы

Примеры

стандартный ввод	стандартный вывод
0 0 1 1 1	2 0.0000000000 1.0000000000 1.0000000000 1.0000000000
0 0 6 8 5	2 3.0000000000 4.0000000000 6.0000000000 8.0000000000

Задача Е. Простые суммы

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Я — простой человек... Делюсь только на единицу и на самого себя.
(Я. Графоманов «Простые люди и числа».)

В соревнованиях по бадминтону участвуют n игроков. Каждый игрок имеет свой индивидуальный номер — целое число от 1 до 10^6 . Перед началом первого тура всех спортсменов разбивают на игровые пары так, чтобы каждый спортсмен участвовал не более, чем в одной паре.

Судья соревнований — большой оригинал, эстет и любитель математики — решил разбить участников соревнования на пары таким образом, чтобы сумма номеров участников в каждой паре была *простым* числом.

Вам необходимо определить наибольшее количество пар, которое можно составить из игроков требуемым образом, и вывести эти пары.

Формат входных данных

В первой строке записано целое число n — количество участников соревнования ($1 \leq n \leq 2500$).

Во второй строке содержатся n различных целых чисел в диапазоне от 1 до 10^6 включительно — номера участников.

Формат выходных данных

В первой строке выведите число k — наибольшее возможное количество пар участников с простой суммой их номеров.

В следующих k строках описание разбиение участников соревнования на пары. Каждая из этих k строк должна содержать два целых числа — номера участников, из которых составлена пара.

Если решений несколько, выведите любое из них.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
1	30	$1 \leq n \leq 10$		баллы
2	30	$1 \leq n \leq 100$	1	баллы
3	20	$1 \leq n \leq 1500$	1, 2	баллы
4	20	$1 \leq n \leq 2500$	1, 2, 3	баллы

Примеры

стандартный ввод	стандартный вывод
4 12 9 6 3	0
7 20 1 2 3 12 13 14	2 20 3 2 1