

Общая информация по задачам олимпиады

Доступ к результатам проверки решений задач во время тура

В течение тура Вы можете просматривать информацию о результатах оценивания решения на тестах жюри. Эта информация будет доступна сразу после проверки Вашего решения на всех тестах. Обратите внимание, что в конце олимпиады система может не успеть проверить Ваше решение до конца соревнования.

Ограничение на размер исходного кода программы-решения

Во всех задачах размер файла с исходным кодом решения не должен превышать 256 КБ.

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. Обратите внимание, что минимальный промежуток времени между послылками для каждой задачи 3 минуты. Это ограничение снимается за 15 минут до конца олимпиады.

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. Билеты	2 секунды	256 МБ	Сообщаются только баллы за пройденные тесты.
В. Сумма факториалов	2 секунды	256 МБ	Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи.
С. Наименьший палиндром	2 секунды	256 МБ	Сообщаются только баллы за пройденные тесты.
Д. Заводы в городе	2 секунды	256 МБ	Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи.
Е. НОК	2 секунды	256 МБ	Для каждой подзадачи сообщаются баллы за эту подзадачу и результат проверки программы на каждом тесте.

Задача А. Билеты

Имя входного файла: `tickets.in`
Имя выходного файла: `tickets.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Группа студентов и школьников собирается в музей. Для школьников до 10 лет (включительно) вход в музей бесплатный, а для остальных школьников и студентов до 18 лет включительно стоимость билета составляет половину от полной стоимости билета. Студенты старше 18 лет покупают билет за полную стоимость — за 100 рублей.

Вам необходимо подсчитать, сколько рублей стоят билеты на всю группу.

Формат входных данных

В первой строке записано целое n — количество студентов и школьников в группе ($1 \leq n \leq 10^6$). Во второй строке записаны n целых чисел, каждое из которых не меньше 7 и не больше 25, — возрасты студентов и школьников.

Формат выходных данных

Выведите одно целое число — суммарную стоимость билетов на всю группу.

Система оценки

Задача оценивается в 100 баллов. Баллы начисляются за каждый пройденный тест.

Примеры

<code>tickets.in</code>	<code>tickets.out</code>
2 9 10	0
3 10 15 20	150

Задача В. Сумма факториалов

Имя входного файла: `factorial.in`
Имя выходного файла: `factorial.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Как известно, факториалом целого положительного числа n называется произведение всех натуральных чисел от 1 до n включительно: $n! = 1 \cdot 2 \cdot \dots \cdot n$. Любое целое положительное число $n > 1$ можно представить в виде суммы факториалов несколькими способами. Например, $3 = 1! + 1! + 1! = 2! + 1!$.

Ваша задача — для данного натурального числа n найти разложение в сумму факториалов с *наименьшим* числом слагаемых. (Например, для числа $n = 10$ такая сумма состоит из трёх факториалов: $3! + 2! + 2!$.)

Формат входных данных

Входные данные содержат одно целое число n ($1 \leq n \leq 10^{18}$).

Формат выходных данных

Выведите одно целое число — наименьшее количество слагаемых-факториалов в разложении данного числа n .

Система оценки

Номер подзадачи	Баллы	Ограничения	Комментарии
		n	
1	50	$1 \leq n \leq 10^9$	Баллы начисляются, если пройдены все тесты.
2	50	$1 \leq n \leq 10^{18}$	Баллы начисляются, если пройдены все тесты этой и предыдущей подзадачи.

Примеры

<code>factorial.in</code>	<code>factorial.out</code>
8	2
10	3

Задача С. Наименьший палиндром

Имя входного файла: `palindrome.in`
Имя выходного файла: `palindrome.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Нажал кабан на баклажан.

Sum sumtus tus.

(Я — сильнейшая мышь.)

Палиндромом будем называть число, запись которого в десятичной системе счисления одинаково читается слева направо и справа налево. Например, числа 131 и 2112 являются палиндромами, а число 2211 — нет. (Напомним, что запись числа не может начинаться с нуля). Палиндромы встречаются не так часто, но иногда их можно сконструировать, переставляя цифры какого-нибудь числа. В частности, переставив цифры в числе 2211, можно получить два палиндрома 1221 и 2112.

Вам необходимо из всех цифр данного числа составить *наименьшее* число-палиндром.

Формат входных данных

В первой строке одно целое n — количество цифр в данном числе ($1 \leq n \leq 10^6$). Во второй строке записано число из n десятичных цифр, первая цифра которого отлична от нуля.

Формат выходных данных

Выведите -1 , если из числа нельзя образовать n -значный палиндром. Иначе выведите наименьшее число-палиндром, которое можно составить из всех цифр данного числа.

Система оценки

Задача оценивается в 100 баллов. Баллы начисляются за каждый пройденный тест.

Примеры

<code>palindrome.in</code>	<code>palindrome.out</code>
4 2211	1221
1 7	7
3 120	-1

Задача D. Заводы в городе

Имя входного файла:	distance.in
Имя выходного файла:	distance.out
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

В 2014 году открылась первая в мире велодорожка из солнечных батарей.

Город Солнечный состоит из k районов, в каждом из которых n домов. Планировка города такая, что все n домов в каждом районе расположены в вершинах выпуклого n -угольника $A_1 A_2 \dots A_n$. Городу требуется электроэнергия, поэтому в каждом из k районов планируется построить свой завод по производству солнечных батарей. Партия местных экологов требует установить каждое предприятие внутри или на границе района так, чтобы сумма расстояний до всех домов района была бы *наибольшей*. В случае необходимости можно построить завод на месте одного из домов района. (Разумеется, переселив для этого всех жильцов этого дома.)

Ваша задача — определить требуемое расположение всех k заводов города.

Формат входных данных

В первых строке записано два целых числа: k — количество районов ($1 \leq k \leq 100$) и n — количество домов в районах города ($3 \leq n \leq 500$). Число домов n одно и то же для всех районов. В каждой из следующих k строк записаны через пробел $2n$ целых чисел — координаты вершин A_1, A_2, \dots, A_n выпуклого многоугольника $A_1 A_2 \dots A_n$, заданные в порядке его обхода по часовой стрелке. Координаты всех вершин — целые числа, не превосходящие по модулю 10^4 . Взаимное расположение районов не имеет значения.

Формат выходных данных

Выходные данные должны содержать k строк. В i -ой строке сначала запишите два числа — координаты искомой точки, для которой сумма расстояний до всех домов i -го района наибольшая, затем ещё одно число — значение этой суммы. Ответ считается правильным, если абсолютная или относительная погрешность не превышает 10^{-5} . Если искомых точек несколько, выведите координаты любой из них.

Система оценки

Номер подзадачи	Баллы	Ограничения	Комментарии
		k, n	
1	25	$1 \leq k \leq 100,$ $n = 3$	Баллы начисляются, если пройдены все тесты этой подзадачи.
2	25	$1 \leq k \leq 100,$ $3 \leq n \leq 20$	Баллы начисляются, если пройдены все тесты этой и предыдущей подзадачи.
3	25	$1 \leq k \leq 100,$ $3 \leq n \leq 300$	Баллы начисляются, если пройдены все тесты этой и предыдущих подзадач.
4	25	$1 \leq k \leq 100,$ $3 \leq n \leq 500$	Баллы начисляются, если пройдены все тесты этой и предыдущих подзадач.

Пример

distance.in	distance.out		
1 3 0 0 0 3 4 0	4.00	0.00	9.000000

Задача Е. НОК

Имя входного файла: `lcm.in`
Имя выходного файла: `lcm.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Двоечник Петька был ленивым и часто прогуливал уроки. Теплым майским днём решил Петька вместо урока математики сходить на речку, через лес. Но не смог он добраться до речки — встретила ему на пути Баба-Яга. Решила она проучить Петьку и не отпускать домой, пока не решит задачу по математике. По той самой теме, которую он прогулял. А задача была такая. Назвала Баба-Яга два натуральных числа m и k . И нужно было Петьке найти количество упорядоченных наборов из k натуральных чисел, у которых наименьшее общее кратное равно данному числу m . Например, для $m = 10$ и $k = 2$ существует 9 наборов из двух целых положительных чисел, у которых наименьшее общее кратное равно 10:

(1; 10), (10; 1), (2; 10), (10; 2), (5; 10), (10; 5), (10; 10), (2; 5), (5; 2).

И теперь вам нужно написать программу, которая позволит дать ответ на задачу и поможет Петьке вернуться домой.

Формат входных данных

Входные данные содержат два целых числа m и k — наименьшее общее кратное и количество чисел в наборах ($1 \leq m \leq 10^9$, $2 \leq k \leq 10^{18}$).

Формат выходных данных

Выведите искомое количество наборов по модулю $(10^9 + 9)$.

Система оценки

Номер подзадачи	Баллы	Ограничения	Комментарии
		m, k	
1	30	$1 \leq m \leq 100,$ $k = 2, 3$	Баллы начисляются, если пройдены все тесты.
2	35	$1 \leq m \leq 10^9,$ $2 \leq k \leq 10^5$	Баллы начисляются, если пройдены все тесты этой и предыдущей подзадачи.
3	35	$1 \leq m \leq 10^9,$ $2 \leq k \leq 10^{18}$	Баллы начисляются, если пройдены все тесты этой и предыдущих подзадач.

Примеры

<code>lcm.in</code>	<code>lcm.out</code>
10 2	9
10 3	49