

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.

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.

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. Конфеты | 2 секунды | 256 МБ | Сообщаются только баллы за пройденные тесты. |
| В. Похожие числа | 2 секунды | 256 МБ | Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи. |
| С. Штатбеля | 1 секунда | 256 МБ | Для каждой подзадачи сообщаются только баллы за пройденные тесты этой подзадачи. |
| Д. Цепочка вычитаний | 2 секунды | 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. Конфеты

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

Карлсон — самый лучший в мире упаковщик конфет!

Если количество конфет в некоторых коробках оказывается разным, Карлсон перекладывает часть конфет так, чтобы во всех коробках было конфет поровну. Если это сделать невозможно, он съедает одну конфету и так поступает до тех пор, пока не наступит момент, когда оставшиеся конфеты можно будет распределить поровну между коробками. Например, если в трёх коробках находится 3, 4 и 5 конфет соответственно, то Карлсон перекладывает одну конфету из последней коробки в первую, при этом во всех коробках будет одинаковое количество конфет — по 4. Однако, если в коробках 3, 4 и 7 конфет, то Карлсон сначала съедает 2 конфеты в последней коробке, а затем перекладывает из неё одну конфету в первую коробку.

Ваша задача — определить *наименьшее* число конфет, которые придётся съесть Карлсону, а также *наименьшее* число конфет, которые ему нужно переложить, чтобы во всех коробках было одинаковое количество конфет.

Формат входных данных

Первая строка содержит одно целое число n — количество коробок с конфетами ($2 \leq n \leq 1000$). Во второй строке записаны через пробел n целых чисел — количества конфет в каждой из коробок. В каждой коробке целое количество конфет — не менее одной и не более 1000.

Формат выходных данных

В ответе запишите через пробел два целых числа: наименьшее количество конфет, которые придётся Карлсону съесть (возможно, ни одной), и наименьшее количество конфет, которые ему нужно переложить.

Система оценки

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

Примеры

| <code>candy.in</code> | <code>candy.out</code> |
|-----------------------|------------------------|
| 3 3 4 5 | 0 1 |
| 3 7 3 4 | 2 1 |

Задача В. Похожие числа

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

Вам даны два числа a и b , состоящие из одинакового количества цифр. *Степенью похожести* будем называть количество совпадающих цифр в десятичной записи a и b . Например, в записи чисел $a = 2233$ и $b = 4232$ есть три совпадающие цифры — две двойки и одна тройка, поэтому их степень похожести равна трём. А вот в записи чисел $a = 100$ и $b = 234$ нет одинаковых цифр, поэтому степень их похожести равна 0.

Вам необходимо составить программу, которая определяет степень похожести двух заданных чисел.

Формат входных данных

В первой строке одно целое n — количество цифр в записи чисел a и b ($1 \leq n \leq 10^6$). Во второй строке — число a , записанное в виде n десятичных цифр, разделённых пробелом. В третьей строке — число b , записанное в виде n десятичных цифр, разделённых пробелом. Первая (слева) цифра у чисел a и b отлична от нуля.

Формат выходных данных

Выведите степень похожести чисел a и b .

Система оценки

| Номер подзадачи | Баллы | Ограничения | Комментарии |
|-----------------|-------|----------------------|---|
| | | n | |
| 1 | 30 | $1 \leq n \leq 9$ | Баллы начисляются, если пройдены все тесты. |
| 2 | 35 | $1 \leq n \leq 10^4$ | Баллы начисляются, если пройдены все тесты этой и предыдущей подзадачи. |
| 3 | 35 | $1 \leq n \leq 10^6$ | Баллы начисляются, если пройдены все тесты этой и предыдущих подзадач. |

Примеры

| <code>similar.in</code> | <code>similar.out</code> |
|-------------------------|--------------------------|
| 3 1 0 0 2 0 0 | 2 |
| 1 5 5 | 1 |

Задача С. Штабеля

| | |
|-------------------------|------------------------|
| Имя входного файла: | <code>stack.in</code> |
| Имя выходного файла: | <code>stack.out</code> |
| Ограничение по времени: | 1 секунда |
| Ограничение по памяти: | 256 мегабайт |

В соответствии с технологической документацией каждый штабель формируется из k коробок путем установки их друг на друга, каждая коробка весит целое число килограммов. Для обеспечения прочности штабеля вес каждой коробки должен быть не меньше суммарного веса всех находящихся над ней коробок. Такой штабель считается *прочным*, и он может быть допущен к эксплуатации. Например, из трёх коробок массой 1, 3 и 5 килограммов можно сформировать прочный штабель: коробку массой 1 кг нужно поставить на верх штабеля, массой 3 кг — второй сверху, и, наконец, коробку массой 5 кг — в низ штабеля.

Вам необходимо составить программу, которая для каждого набора из k коробок определяет, можно ли из них сформировать прочный штабель.

Формат входных данных

В первой строке записано два целых числа: n — количество штабелей и k — количество коробок в каждом штабеле ($1 \leq n \leq 20\,000$; $2 \leq k \leq 50$). В каждой из следующих n строк записаны разделенные пробелом k целых чисел — массы коробок в каждом из n штабелей. Каждая коробка весит целое число килограммов, не менее 1 кг и не более 10^{18} кг.

Формат выходных данных

Выходные данные содержат n строк. В i -й строке должно быть записано `yes`, если из k коробок i -го набора можно сформировать прочный штабель, и `no` — в противном случае.

Система оценки

| Номер подзадачи | Баллы | Ограничения | Комментарии |
|-----------------|-------|--|---|
| | | n, k | |
| 1 | 30 | $1 \leq n \leq 10,$ $2 \leq k \leq 50$ | Баллы начисляются, если пройдены все тесты этой подзадачи. |
| 2 | 35 | $1 \leq n \leq 5\,000,$ $2 \leq k \leq 50$ | Баллы начисляются, если пройдены все тесты этой и предыдущей подзадачи. |
| 3 | 35 | $1 \leq n \leq 20\,000,$ $2 \leq k \leq 50$ | Баллы начисляются, если пройдены все тесты этой и предыдущих подзадач. |

Примеры

| <code>stack.in</code> | <code>stack.out</code> |
|-----------------------|------------------------|
| 1 3 4 1 5 | yes |
| 2 3 5 1 3 5 3 3 | yes no |

Задача D. Цепочка вычитаний

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

На доске записаны $n + 1$ степеней двоек: $1, 2^1, 2^2, 2^3, \dots, 2^n$. Разрешается стереть любые два числа и вместо них записать их разность — *неотрицательное* число. После n таких операций на доске останется только одно число.

Вам необходимо найти такую последовательность операций вычитания, с помощью которой из исходных чисел получается данное число k .

Формат входных данных

В единственной строке — два целых числа n и k ($1 \leq n \leq 60; 1 \leq k \leq 10^{18}$). Первое число n означает, что на доске первоначально записаны $n + 1$ чисел: $1, 2^1, 2^2, 2^3, \dots, 2^n$. Второе задаёт число, которое необходимо получить.

Формат выходных данных

Если число k получить можно, выведите n строк. В i -й строке запишите два числа, которые стираются на i -м шаге. (Вместо них появится разность стёртых чисел. Это новое число можно использовать на одном из следующих шагов.) Если решений несколько, выведите любое из них. Выведите -1 , если число k получить невозможно.

Система оценки

| Номер подзадачи | Баллы | Ограничения | Комментарии |
|-----------------|-------|--|---|
| | | n, k | |
| 1 | 30 | $1 \leq n \leq 10,$ $1 \leq k \leq 10$ | Баллы начисляются, если пройдены все тесты. |
| 2 | 35 | $1 \leq n \leq 30,$ $1 \leq k \leq 10^9$ | Баллы начисляются, если пройдены все тесты этой и предыдущей подзадачи. |
| 3 | 35 | $1 \leq n \leq 60,$ $1 \leq k \leq 10^{18}$ | Баллы начисляются, если пройдены все тесты этой и предыдущих подзадач. |

Примеры

| | <code>subtract.in</code> | <code>subtract.out</code> |
|--|--------------------------|---------------------------|
| | 3 5 | 1 2 4 1 8 3 |
| | 3 4 | -1 |

Замечание

В первом примере $n = 3$, и число $k = 5$ получается из последовательности $1, 2^1, 2^2, 2^3$ с помощью цепочки из $n = 3$ вычитаний: $5 = 8 - (4 - (2 - 1))$, то есть сначала составляем разность чисел 2 и 1 (остаётся $2 - 1 = 1$), затем — разность 4 и полученного числа 1 (остаётся $4 - 1 = 3$), наконец, стираем два оставшихся числа 8 и 3, заменив их разностью $8 - 3 = 5$.

Во втором примере ответ -1 , так как число $k = 4$ нельзя получить из того же набора чисел $1, 2^1, 2^2, 2^3$ с помощью $n = 3$ вычитаний.