

Задача А. Усердные бобры

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

Бобры — уникальные животные. Больше всего они известны своим умением строить дамбы на реках.

Рассмотрим подробно процесс постройки дамбы. Исходно бобер стоит перед длинным рядом деревьев. Будем считать этот ряд бесконечным в обе стороны. Каждое дерево в ряду может быть либо *хорошим*, либо *плохим*. Исходно все деревья хорошие. Но после того, как бобер погрызет хорошее дерево, оно может стать плохим, и наоборот, после того как бобер погрызет плохое дерево, оно может стать хорошим.

Бобер может быть в одном из пяти настроений: он может быть сердитым (*angry*), усердным (*busy*), творческим (*creative*), отчаявшимся (*despaired*) и утомленным (*exhausted*). Исходно бобер сердит.

Когда бобер стоит перед деревом, в зависимости от своего настроения и того, хорошее ли дерево перед ним, он делает следующие действия:

- грызет дерево, при этом возможно оно меняется с плохого на хорошее или наоборот;
- меняет свое настроение на одно из возможных пяти (настроение бобра может остаться тем же);
- перемещается либо к соседнему дереву налево, либо к соседнему дереву направо.

Также при некоторой комбинации настроения бобра и состояния дерева перед ним, бобер может наконец решить, что плотина готова, и становится счастливым (*happy*). Может случиться, однако, что бобер так никогда и не станет счастливым.

Вам заданы правила, по которым действует бобер. Выясните, станет ли этот бобер счастливым.

Формат входных данных

Входной файл содержит десять слов, разделенных пробелами. Каждое слово состоит из трех символов, эти слова задают поведение бобра. Первое слово задает поведение сердитого бобра на хорошем дереве, второе слово задает поведение сердитого бобра на плохом дереве, третье слово задает поведение усердного бобра на хорошем дереве, и т.д., последнее слово задает поведение утомленного бобра на плохом дереве.

Первый символ каждого слова равен «1», если бобер делает дерево плохим, либо «0», если бобер делает дерево хорошим. Второй символ равен «R», если бобер перемещается направо, либо «L», если бобер перемещается налево. Наконец, третий символ равен либо букве от «A» до «E» и описывает новое настроение бобра, либо «H», если бобер становится счастливым.

Формат выходных данных

Выведите «happy beaver», если бобер становится счастливым, или «unhappy beaver», если бобер не становится счастливым.

Примеры

стандартный ввод	стандартный вывод
1RB 1LC 1RC 1RB 1RD OLE 1LA 1LD 1RH OLA	happy beaver
ORA ORA ORA ORA ORA ORA ORA ORA ORA ORA	unhappy beaver

Задача В. Карта

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

Хулиган Вася получил двойку по географии и начал от злости рвать карту. Делал он это в течение t секунд.

Каждую секунду он делал следующее: брал первый попавшийся кусок карты и разрывал его на k частей. Окончив рвать карту, Вася обнаружил перед собой n обрывков карты.

Пока Вася уничтожал карту, он потерял счет времени и теперь ему интересно, как долго он рвал карту, поэтому он попросил Вас помочь ему найти t .

Формат входных данных

В первой строке даны два целых числа n и k ($1 \leq n, k \leq 10^9$).

Формат выходных данных

Если не существует подходящего t , то выведите единственное число -1 , иначе выведите t .

Примеры

стандартный ввод	стандартный вывод
10 4	3
10 3	-1

Задача С. Сумма степеней

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

Мальчик Сережа очень любит изучать числовые последовательности. Он уже изучил все числовые последовательности, встречающиеся в книгах, и теперь просит всех своих знакомых придумать ему последовательность. Также он попросил придумать числовую последовательность и своего младшего брата Ромочку, который умеет считать только до шести. Разумеется, Ромочка (после долгих рассказов Сережи о том, что такое последовательность) предложил Сереже изучить последовательность 1, 2, 3, 4, 5, 6. Но Сережа уже неоднократно изучал эту последовательность, и немного изменил её. Теперь он изучает последовательность натуральных чисел, такую, что если число k её член, то k является делителем числа $1 + 2^k + 3^k + 4^k + 5^k + 6^k$.

Например, первым числом в этой последовательности является 1, так как $1 + 2 + 3 + 4 + 5 + 6 = 21$ и это число делится на единицу. Двойка в эту последовательность не входит, так как $1 + 4 + 9 + 16 + 25 + 36 = 91$ не делится на 2. Число 3 в эту последовательность также входит, так как $1 + 8 + 27 + 64 + 125 + 216 = 441$ и делится на 3.

Сережа вычислил несколько первых членов последовательности, после чего решил, что лучше напишет программу, которая за него посчитает член последовательности с номером n . Напишите и Вы такую программу.

Формат входных данных

Входной файл содержит единственное число n ($1 \leq n \leq 65$).

Формат выходных данных

В выходной файл выведите искомый член последовательности.

Примеры

стандартный ввод	стандартный вывод
1	1
2	3

Задача D. Строки Фибоначчи

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

В математике достаточно часто применяются так называемые рекуррентные соотношения. Обычно они применяются для задания числовых последовательностей, но могут применяться и для задания последовательностей строк.

Одним из примеров строк, задаваемых рекуррентным соотношением являются строки Фибоначчи F_0, F_1, \dots . Они задаются следующим образом: $F_0 = a, F_1 = b, F_i = F_{i-2}F_{i-1}, i > 1$. Первые семь строк Фибоначчи выглядят следующим образом: a, b, ab, bab, abbab, bababbab, abbabbababbab.

Дима занимается в кружке олимпиадного программирования и интересуется алгоритмами на строках. Недавно он узнал о строках Фибоначчи. Он быстро понял, что их длина с увеличением номера i растет очень быстро, поэтому задача нахождения всех символов строки F_i требует слишком большого объема памяти. Поэтому он решил ограничиться задачей нахождения некоторых символов.

Напишите программу, которая находит k -ый символ строки F_i .

Формат входных данных

Входной файл содержит несколько наборов входных данных. Первая строка входного файла содержит целое число T наборов входных данных ($1 \leq T \leq 100$). Каждая из последующих T строк описывает один набор входных данных и содержит по два целых числа: n и k ($0 \leq n \leq 45, 1 \leq k \leq |F_n|$, как $|F_n|$ обозначена длина строки F_n , позиции символов в строке нумеруются с единицы).

Формат выходных данных

Выведите в выходной файл T строк, каждая из которых должна содержать ровно один символ — ответ для соответствующего набора входных данных.

Пример

стандартный ввод	стандартный вывод
4	a
0 1	b
1 1	a
3 2	a
7 7	

Задача E. Дробные фокусы

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

Мальчик Сережа очень хочет научиться жонглировать и показывать фокусы. Недавно его друг показал ему один фокус — попросил Сережу загадать некоторую положительную дробь $\frac{a}{b}$, затем назвать ее знаменатель и две дроби $\frac{p_1}{q_1}$ и $\frac{p_2}{q_2}$, такие, что $\frac{p_1}{q_1} < \frac{a}{b} < \frac{p_2}{q_2}$, причем между ними нет меньшей дроби с тем же знаменателем, что и у дроби, загаданной Сережей. После этого Сережин друг отгадал числитель загаданной Сережей дроби.

Теперь Сережа хочет научиться сам показывать этот фокус. Помогите ему в этом — напишите программу, находящую числитель загаданной дроби.

Формат входных данных

Во входном файле через переводы строк заданы три дроби в формате <числитель>/<знаменатель> ($p_1/q_1, a/b, p_2/q_2$), где вместо числа a указан вопросительный знак «?». Все числители и знаменатели являются положительными целыми числами и не превышают 1000.

Формат выходных данных

В выходной файл выведите минимальное число a , такое, что $\frac{p_1}{q_1} < \frac{a}{b} < \frac{p_2}{q_2}$. Если такого не существует, выведите «-1» (без кавычек).

Примеры

стандартный ввод	стандартный вывод
1/2 ?/8 3/4	5/8
3/20 ?/2 4/10	-1

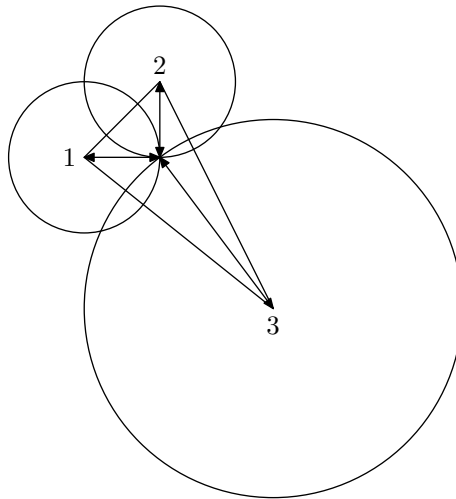
Задача F. Система глобальнейшего позиционирования

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

Говорят, если в условии задачи нарисовать 3 окружности, люди даже не будут пытаться её читать.

Недавно во Флатландии было решено создать Новейшую Систему Глобальнейшего Позиционирования. Поскольку страна занимает бесконечно большой участок плоскости, то вывод спутников очень затруднителен, поэтому было решено ограничиться наземным методом позиционирования.

Для этого во Флатландии было построено три радиовышки, не находящиеся на одной прямой. Объект, который хочет узнать свое местоположение, посылает вышкам сигнал. По силе сигнала, дошедшего до вышек, определяется расстояние между вышками и объектом.



Напишите программу, которая реализует последний компонент системы, который, получая координаты вышек и расстояния от объекта до каждой из них, находит координаты объекта.

Формат входных данных

В первой строке входного файла три пары чисел x_1, y_1, x_2, y_2, x_3 и y_3 — координаты вышек. Во второй строке три неотрицательных числа — расстояния до соответствующих вышек. Все числа во входном файле целые и по модулю не превышают 50.

Формат выходных данных

Если не существует такого местоположения объекта, что расстояния до вышек соответствовали бы данным, то выведите в выходной файл единственное слово «Impossible». Иначе выведите два числа — координаты объекта. Ответ будет проверяться с точностью до шести знаков после запятой.

Примеры

стандартный ввод	стандартный вывод
0 4 2 6 5 0 2 2 5	2.0 4.0
0 0 0 3 1 -4 4 5 5	4.0 0.0
0 0 1 0 0 1 2 2 2	Impossible

Задача G. Перестановка букв

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

По результатам исследования одонго анлигйсокго унвиертисета, не имеет значения, в каком порядке расположены буквы в слове. Галвоне, чтобы преавя и слоендя буквы были на месте. Остальные буквы могут следовать в любом беспорядке, все равно текст читается без проблем. Причиной этого является то, что мы не читаем каждую букву по отдельности, а все слово целиком.

Ваша задача — подтвердить исследования английских ученых для произвольного текста.

Каждое слово s в тексте, если это возможно, необходимо преобразовать так, чтобы первая и последняя буква остались на своих местах, а остальные буквы стояли вперемешку, но так, чтобы преобразованное слово не совпадало с исходным. Если же слово указанным образом преобразовать невозможно, то его следует оставить без изменений.

Напишите программу, выполняющую указанное преобразование.

Формат входных данных

Во входном файле задан текст, состоящий из заглавных и строчных латинских букв, знаков препинания: «!?:;-.» и пробелов. Словом считается любая последовательность подряд идущих букв, ограниченная пробелами, знаками препинания, переводом строки, началом или концом текста. Длина текста не превышает 10 килобайт, длина каждого слова не превышает 20 символов. Последняя строка файла завершается переводом строки.

Формат выходных данных

В выходной файл выведите тот же самый текст, каждое слово в котором преобразовано согласно условию. В примере, для пояснения, пробелы заменены подчеркиваниями. Ваша программа должна выводить пробелы.

Примеры

стандартный ввод	стандартный вывод
Hello_World!	Hlleo_Wolrd!
Po__rezultatam...	Po__rzeluatatm...
QaQ_QaaaaQ	QaQ_QaaaaQ

Задача N. Системы счисления

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

Чак Норрис досчитал до бесконечности. Дважды.

Интернет-фольклор

Юный математик Вовочка столь неленив, что выписал в ряд все натуральные числа в порядке возрастания. В порыве трудолюбия он перевел все числа ряда в систему счисления с основанием b . Потом Вовочка вычеркнул все числа ряда, которые являются записью какого-либо числа в системе счисления с основанием c .

Трудолюбие Вовочки закончилось, и он попросил Вас, как своего друга-программиста, найти n -ое невычеркнутое число ряда. Помогите юному математику Вовочке!

Формат входных данных

Единственная строка входного файла содержит три числа n , b и c ($1 \leq n \leq 10^7$, $2 \leq c < b \leq 36$). В системах счисления с основаниями, большими 10 в качестве цифр кроме $0, \dots, 9$ используются также строчные буквы латинского алфавита a, \dots, z .

Формат выходных данных

В выходной файл выведите (в десятичной системе счисления) n -ое невычеркнутое число ряда.

Примеры

стандартный ввод	стандартный вывод
2 3 2	5
1 3 2	2
1 36 35	35

Задача I. Кодовый замок

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

Вася очень любит использовать кодовые замки. Как известно, кодовый замок состоит из барабана с несколькими кольцами. Причем, у каждого кольца есть несколько положений, каждое из которых соответствует некоторой цифре в k -ичной системе счисления. Таким образом, любое состояние всего кодового замка с n кольцами можно представить некоторым n -значным числом в k -ичной системе счисления. При этом замок открывается только в одном фиксированном состоянии.

Так как код замка очень легко забыть, для его запоминания Вася пользуется хитрым способом. После закрытия замка Вася поворачивает замок таким образом, чтобы число, кодирующее его состояние, было предыдущим (в порядке возрастания) числом, имеющим ровно такую же сумму цифр, что и то число, при котором замок открывается.

Тогда для открытия замка Васе достаточно найти следующее (в порядке возрастания) число с той же суммой цифр в k -ичной системе счисления, что и текущее, и это число откроет замок.

Но, после месяца использования такого замка, Васе надоело каждый раз решать вручную такую непростую задачу, и он попросил Вас помочь ему.

Требуется написать программу, которая будет по заданному числу m в k -ичной системе счисления находить следующее число (в порядке возрастания) в этой же системе счисления с такой же суммой цифр.

Формат входных данных

В первой строке входного файла задано два натуральных числа k и n ($2 \leq k \leq 36$; $n \leq 100\,000$). Во второй строке задано число m в k -ичной системе счисления, при этом число m состоит из n цифр и, возможно, содержит ведущие нули. Цифрам от 10 до 35 соответствуют соответственно заглавные латинские буквы от A до Z .

Формат выходных данных

В выходной файл выведите ответ на задачу — искомое число в k -ичной системе счисления из n цифр s , возможно, несколькими ведущими нулями. Если Вася где-то ошибся, и следующего такого числа не существует, то выведите в выходной файл единственное слово «Impossible».

Примеры

стандартный ввод	стандартный вывод
10 2 23	32
16 2 FF	Impossible
26 2 HP	I0

Задача J. MP3-плеер

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

Вася недавно купил себе новый MP3-плеер doPi Shuffle. Этот плеер отличается от всех других, которые использовал Вася, тем, что у него есть функция «перемешивания» списка проигрываемых песен (плейлиста).

Вася загрузил в плеер n своих любимых песен. До этого он их постоянно слушал на компьютере и очень хорошо запомнил порядок, в котором они были в плейлисте.

Разумеется, плеер переставил песни в случайном порядке. Прослушав плейлист несколько раз на плеере, Вася с удивлением обнаружил, что никакие две, шедшие подряд друг за другом в старом плейлисте, не идут подряд в новом. Он задумался, насколько часто возможна такая ситуация.

Пронумеруем песни числами от 1 до n в том порядке, в котором они идут в плейлисте на компьютере. Напишите программу, которая вычисляет число способов их переставить так, чтобы никакие две песни с последовательными номерами не шли непосредственно друг за другом. Так как это число может быть достаточно большим, выведите его по модулю m .

Формат входных данных

Входной файл содержит два целых числа: n и m ($2 \leq n \leq 2000$, $2 \leq m \leq 10^9$).

Формат выходных данных

В выходной файл выведите остаток от деления числа способов переставить песни требуемым образом на m .

Примеры

стандартный ввод	стандартный вывод
2 1000000	1
5 1000000	53