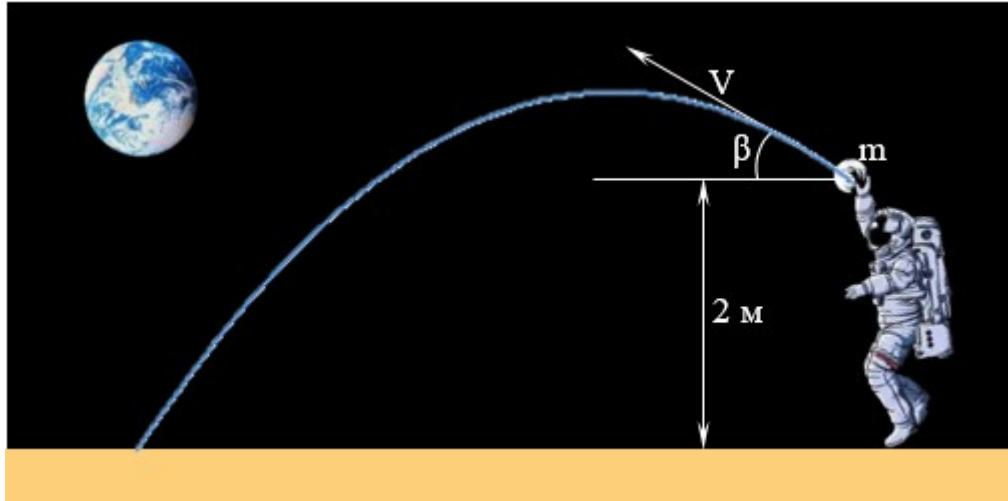


А. Астронавт

Ограничение по времени: **1000 миллисекунд**
Ограничение по памяти: **65000 кибибайт**

Астронавт Василий, находясь на Луне, кинул камень массой m (кг) с начальной скоростью V (м/с) под углом β (в градусах) к её поверхности. Необходимо определить длину пройденного камнем пути S (в метрах) до момента столкновения его с поверхностью Луны.



При расчётах считайте, что в момент броска камень находится на высоте 2-х метров над поверхностью Луны, атмосферы на Луне нет (трение об атмосферу отсутствует), камень является материальной точкой, поверхность Луны – плоскость (без ям и возвышенностей), притяжение над всей поверхностью Луны постоянно, не зависит от высоты и ускорение свободного падения составляет $1/6$ от ускорения свободного падения на Земле ($9,8 \text{ м/с}^2$).

Входные данные

Три целых числа разделённых пробелом: m ($1 \leq m \leq 100$), V ($0 \leq V \leq 10$), β ($-90 \leq \beta \leq 90$).

Выходные данные

Одно целое число S – длина пути в метрах. Ответ будет считаться верным, если ошибка от точного значения составит менее одного метра.

Примеры

Ввод	Вывод
100 0 0	2
1 10 -90	2

В. Вода

Ограничение по времени: **1000 миллисекунд**
Ограничение по памяти: **65000 кибибайт**

Василий живёт в деревне и ему время от времени нужно по хозяйству наполнить водой ёмкость (до краёв), представляющую собой прямоугольный параллелепипед со сторонами: L (длина), W (ширина) и H (высота). Для переноса воды из колодца в ёмкость он будет использовать ведро, имеющее форму усечённого конуса с диаметрами оснований d_1 (дно ведра) и d_2 (верх ведра) и высотой h .

Что бы оценить количество времени, которое Василий потратит на заполнение ёмкости и назначить встречу, на которую он придёт вовремя, ему нужно оценить, какое минимальное количество полных (до краёв) вёдер ему необходимо будет принести, чтобы заполнить ёмкость.

Помогите Василию и другим ребятам (в каждом доме в хозяйстве имеются различные по габаритам ёмкости и вёдра), напишите программу, которая по заданным L , W , H , d_1 , d_2 , h будет находить минимальное количество полных вёдер воды, которые нужно принести, чтобы до краёв заполнить ёмкость.

Входные данные

В первой строке через пробел даны целые числа L , W , H ($1 \leq L, W, H \leq 10^3$).

Во второй строке через пробел даны целые числа d_1 , d_2 , h ($1 \leq d_1, d_2, h \leq 10^3$).

Выходные данные

Одно целое число – ответ на задачу.

Пример

Ввод	Вывод
10 10 10	2
10 10 10	

С. Мат в один ход

Ограничение по времени: **1000 миллисекунд**
Ограничение по памяти: **65000 кибибайт**

Юлия – юный шахматист дошкольного возраста, она только учится играть в эту древнюю игру, на каждой неделе дважды посещает кружок и быстро прогрессирует. В кружке их обучает Григорий Сергеевич. На первых занятиях им объяснили, какова цель игры и как ходят фигуры.

Сейчас они учатся решать задачи, где нужно поставить мат в один ход. Так как некоторые правила игры немного сложны для начинающих, Григорий Сергеевич даёт только такие задачи, где для решения не нужно знать что такое «рокировка», «превращение пешки в фигуру» и «взятие на проходе».

Давайте поможем юным шахматистам и создадим сервис, который умел бы проверять правильность решения задач. Один из модулей системы должен по описанию позиции находить мат в один ход. Напишите такую программу.

Входные данные

Восемь строк, описывающих линию за линией шахматное поле (см. примеры задач). Гарантируется, что каждая строка заканчивается переводом строки и не содержит пробелов. Для обозначения пустой клетки используется знак «#» (ASCII код 35). Во всех позициях белые начинают и дают мат в один ход. Белые пешки идут снизу вверх, чёрные сверху вниз, то есть первая горизонталь внизу, восьмая сверху.

Обозначение фигур и пешек осуществлено латинскими буквами:

- белые фигуры: P – пешка, N – конь, B – слон, R – ладья, Q – ферзь, K – король;
- черные фигуры: p – пешка, n – конь, b – слон, r – ладья, q – ферзь, k – король.

Выходные данные

В том же формате, что и входные данные, вывести позицию с решением.

Примеры

Ввод	Вывод
##### ##### ####B### ##### ###p#### ###pk#N #####Q #K#####	##### ##### ####B### #####N# ###p#### ###pk## #####Q #K#####
r####rk# #####pPp ##### p##### #q####R# ##### #####PP# #####K#	r####rk# #####p#p ##### p##### #q#B##R# ##### #####PP# #####K#

D. Счастливые билеты

Ограничение по времени: **1000 миллисекунд**
Ограничение по памяти: **65000 кибибайт**

«Счастливые билеты» – это такие билеты, номера которых имеют чётное количество цифр, и сумма половины цифр слева равна сумме половины цифр справа.

Иван очень добрый парень, и он, видя, как расстраивается его младшая сестрёнка, когда он приносит «несчастливый билет», который он купил при проезде в автобусе, решил исправлять номер так, чтобы «несчастливые билеты» становились «счастливыми».

Он так качественно исправляет номера, что его сестрёнка не может заметить, и она радуется каждый день новым «счастливым билетикам».

Дело в том, что на исправление одной цифры уходит не менее получаса, помогите Василию – напишите программу, которая будет указывать, какое минимальное количество цифр ему нужно будет исправить.

Входные данные

Номер из чётного количества (не более 100) цифр, возможно с ведущими нулями.

Выходные данные

Одно число – минимальное количество исправлений.

Примеры

Ввод	Вывод
123600	0
123500	1
2299	2

Е. PHP

Ограничение по времени: **1000 миллисекунд**
Ограничение по памяти: **65000 кибибайт**

У Василия есть друг Герман, который занимается созданием сайтов. Василий хочет помочь Герману в этом нелёгком труде, и он обратился к сообществу программистов с просьбой помочь.

Входные данные

Первая строка длиной до 5000 символов, состоящая из латинских букв (строчные и прописные), цифр, знаков препинания, пробелов и другими символами, коды которых в ASCII больше или равны 32. Гарантируется, что строка не начинается и не заканчивается пробелами.

В строке есть переменные – строки в формате %NAME%, где NAME – строка, состоящая из заглавных латинских букв и цифр без пробелов длиной до 10 символов, и начинающаяся всегда с буквы. Для того, чтобы вывести знак процента, он всегда повторяется два раза, это необходимо, чтобы была возможность в результирующей строке получить слово из прописных букв окружённое знаками %.

Далее идёт не более 100 строк в формате NAME=value, где NAME – имя переменной, value – её значение (любая последовательность знаков с кодами большими или равными 32, последовательность знаков не начинающаяся и не заканчивающаяся пробелом). Гарантируется, что значения всех переменных из исходной строки будут определены. Значение каждой переменной определяется только один раз.

Выходные данные

Код программы на PHP, который после исполнения выведет первую строку входных данных с подставленными вместо %NAME% значениями переменных.

Примеры

Ввод	Вывод
Hello, %MYNAME%. Value=100% MYNAME=Vasya	<?php echo 'Hello, Vasya. Value=100%. '; ?>
Hello, %MYNAME%. Value=100% MYNAME=Vasya	<?='Hello, Vasya. Value=100%. ' ?>
My friends: %FRIEND1%, %FRIEND2%. FRIEND1=Ivanov Petr FRIEND2=Stepanov Roman	<?php echo "My friends: "; echo "Ivanov Petr"; echo ", "; echo "Stepanov Roman"; echo " .";
Variable %%%VAR%%=%VALUE% VAR=VALUE VALUE=10%	<?php echo "Variable %VALUE %=10%";
Percents: %N1%, %N2%, %N1%. N1=N2	<?php echo "Percents: " . "N2" . ", " . "N1" . ", " .

N2=N1	"N2" . ". ";
-------	--------------

Примечание:

Вы можете выводить любой PНР код, который после выполнения даст первую строку с подставленными значениями переменных (см. первый и второй примеры с разными выводами). Во время соревнования выдаваемый вашей программой вывод будет интерпретироваться PНР 5.6.30. Код не должен использовать функции для работы с файлами или сетью.

Ф. Взлом

Ограничение по времени: **1000 миллисекунд**
Ограничение по памяти: **65000 кибибайт**

Один из популярных сервисов арифметических расчётов, которым ученики активно пользуются при сдаче ЕГЭ, решил провести следующий опыт на внимательность: когда поступает арифметическое выражение (операции сложения, вычитания и умножения над двумя аргументами), то сервис вместо выдачи правильного ответа будет выдавать другой, состоящий из тех же цифр, что имеются в правильном ответе, но важным является то, что такой результат можно было бы получить при той же операции, пусть и при других целочисленных аргументах любой величины.

Напишите программу, которая позволит провести опыт.

Входные данные

Арифметическое действие над двумя целыми неотрицательными числами (числа не превышают 10^6), формат данных – смотри примеры, арифметический знак с каждой из сторон обрамлён одним пробелом.

Выходные данные

Одно число – неправильный ответ. Если вывести неправильный ответ, отвечающий всем условиям невозможно, то выведите правильный ответ. Ответы должны быть без ведущих нулей, у положительных чисел не должно быть знака «+», ноль так же должен быть без знака.

Пример

Ввод	Вывод
1 + 9	10
9 - 15	-6
19 + 8	72
3 * 7	12

Г. Ноты

Ограничение по времени: **1000 миллисекунд**
Ограничение по памяти: **65000 кибибайт**

Музыкальные звуки, частота которых отличается в два раза, воспринимаются на слух как очень похожие, как повторение одного звука на разной высоте. Это явление называется октавным сходством звуков. На основе этого весь диапазон частот используемых в музыке звуков делится на участки, называемые октавами, при этом частота звуков в каждой последующей октаве будет в два раза выше, чем в предыдущей, а схожие звуки получают одинаковые названия ступеней.

Расположение частотных границ октав условно и выбрано таким образом, чтобы каждая октава начиналась с первой ступени («До») равномерно темперированного двенадцатизвукового строя и при этом частота 6-й ступени («Ля») одной из октав (называемой «первой») составляла бы 440 Гц.

Ваша задача, получив массив отсчётов числовых значений, соответствующих мгновенным значениям амплитуд однотонального сигнала, полученных с помощью аналого-цифрового преобразования с частотой дискретизации 8 кГц, указать частоту этого сигнала.

Входные данные

Восемь тысяч целых чисел по модулю не превосходящих 1024. Данные получены математическим путём по формуле $1024 \cdot \sin(2 \cdot 3,1415/8000 \cdot i \cdot \text{freq})$, дробная часть отбрасывалась, i – номер отсчёта от 0 до 7999, freq – целое число – частота однотонального сигнала ($300 \leq \text{freq} \leq 3400$).

Выходные данные

Одно целое число – частота однотонального сигнала, то есть значение freq .

Пример

Ввод	Вывод
0 464 -828 1011 -973 723 -315 -160 602 -912 1023 -911 601 -159 -317 724 -974 1011 -827 463 1 -466 829 -1011 973 - 722 314 162 -603 913 -1023 911 -599 157 319 -726 974 - 1010 826 -462 -3 467 -830 1011 -972 721 -312 -163 605 - 914 1023 -910 598 -155 -320 727 -975 1010 -825 460 4 -469 831 -1012 972 -720 311 165 -606 914 -1023 909 -597 154 322 -728 975 -1010 824 -459 -6 470 -832 1012 -971 719 - 309 -167 607 -915 1023 -909 595 -152 -323 729 -976 1010 -823 457	3400

Примечание:

В примере приведены только первые 100 значений, во входных данных всегда предоставлены 8000 значений. Переводы строк в примере добавлены для читаемости, во входных данных их нет.