

Задача А. Автоматизация

Time limit: 1 с.

Memory limit: 256 МВ

Для автоматизации процесса сборки танков нового поколения был построен конвейер, для работы которого достаточно было просто нажать кнопку “Старт” и подождать, пока готовый танк съедет с ленты.

К сожалению, первый конвейер был прототипом, поэтому содержал некоторый изъян: в определенный момент времени лента конвейера останавливалась и сборка прекращалась. Причина проблемы была в неправильном размере зубчиков у трех первых из четырех идущих подряд шестеренок: на каждой из шестеренок ровно по одному зубчику были несколько больше, чем все остальные. Лента останавливалась, как только все три зубца попадали в пазы другой шестеренки. После того, как это происходило, требовалось некоторое время на перезапуск конвейера, при этом не нужно было проворачивать шестеренки – они вполне нормально продолжали крутиться из положения, в котором застопорились.

Новые конвейеры были лишены этого дефекта и работали как часы, но утилизировать этот конвейер не стали – вместо этого вас попросили посчитать, через сколько конвейер вновь остановится после того, как его перезапустят.

Входные данные.

В единственной строке ввода содержатся четыре целых положительных числа — количество зубцов у первой, второй, третьей и четвертой шестеренок соответственно. Все числа целые, положительные и не превосходят 1000000. Для упрощения можно предположить, что каждая шестеренка сдвигается на один зубчик за одну секунду.

Выходные данные.

В единственной строке вывода должно быть написано время, через которое шестеренки остановятся вновь после своей первой остановки в формате: x h y min z sec, где x , y , z — часы, минуты и секунды соответственно. Если $x=0$, то выводить x h не нужно, если $x=0$ и $y=0$, то выводить x h y min не нужно. В одной минуте 60 секунд, в одном часе 60 минут.

Примеры.

Входные данные	Выходные данные
6 7 6 6	42 sec

Задача В. Большие учения

Time limit: 1 с.

Memory limit: 256 МВ

В ежегодные сухопутные учения в этом году вошел марш-бросок через минное поле. Поле имеет форму прямоугольника, разбитого на несколько ячеек в форме других, равных друг другу, прямоугольников – так называемых подсекторов, нумерация которых для удобства начинается с нуля. Кроме того, есть некоторые правила, по которым можно перемещаться между подсекторами: из подсектора с координатами $[i;j]$ (первая координата отвечает за номер строки, вторая – за номер столбца) можно переместиться только в подсектор с координатами $[i;j+1]$, $[i+1; j+1]$ или $[i-1; j+1]$. При этом из подсектора, у которого первая координата равна нулю, можно переместиться в подсектор с максимальной возможной первой координатой (для этого специально можно вызвать вертолет) и наоборот. Кроме того, есть одно условие, которое необходимо обязательно выполнить для успешного прохождения марш-броска: во всех секторах, в которых побывает группа, должны быть обезврежены все мины, которые там установлены. Согласно легенде, вы знаете количество мин, расположенных в каждом секторе. Исходная точка, с которой ваша группа может начать – любой подсектор с нулевой второй координатой, конечная – любой подсектор с максимальной второй координатой. Ваша цель – проложить маршрут так, чтобы по пути пришлось обезвредить наименьшее число мин.

Входные данные.

В первой строке даются значения n и m — количество подсекторов, образующих прямоугольник минного поля ($1 \leq n, m \leq 100$). Далее идет n строк, содержащих по m целых положительных чисел — количество мин в соответствующем подсекторе. В i -ой строке j -ым числом является число мин в подсекторе $[i;j]$. Число мин в каждом подсекторе не превосходит 10000.

Выходные данные.

В единственной строке выведите минимальное число мин, которое нужно обезвредить, чтобы успешно завершить прохождение марш-броска.

Пример.

Входные данные	Выходные данные
5 6 3 4 1 2 8 6 6 1 8 2 7 4 5 9 3 9 9 5 8 4 1 3 2 6 3 7 2 1 2 3	11

Подсказка: минимальное число мин придется обезвредить, если идти через подсектора с координатами $[0;0]$, $[1;1]$, $[0;2]$, $[4;3]$, $[3;4]$, $[4;5]$.

Задача С. Скобки
Time limit: 1 с.
Memory limit: 256 МВ

С целью повышения качества текстов военных приказов и уставов, Министерство обороны потребовало реализовать программный комплекс, позволяющий проверять правила орфографии и пунктуации в новом тексте. Работу разбили на группы так, что каждая из групп занимается определенным набором правил. Ваша группа занимается проверкой корректности расстановки скобок в тексте и проверкой правильности их вложенности на различных уровнях. Для удобства в тексте оставили только скобки, все буквы, цифры и знаки препинания были удалены. Ваша задача – определить, является ли скобочная последовательность правильной, то есть узнать, существует ли какой-нибудь уровень вложенности, на котором открывающая и закрывающая скобки отличаются друг от друга.

Входные данные.

В первой строке дано число N — общее количество скобок ($1 \leq N \leq 100000$). Затем дана строка из N символов, в которой содержатся лишь символы “(”, “)”, “{”, “}”, “[”, “]”.

Выходные данные.

В единственной строке должно содержаться слово «Yes», если последовательность правильная, и «No» в противном случае. Ответ выводить без кавычек.

Примеры.

Входные данные	Выходные данные
6 ([()])	No
24 {[O([{}])][{}{}]}[]	Yes

Задача D. Дешифровка

Time limit: 1 с.

Memory limit: 256 МВ

Военные аналитики придумали новый шифр для передачи текстовых сообщений, содержащих информацию о количестве единиц вражеской техники в некоторой стратегической области. Программное обеспечение для шифровки и безопасной передачи данных уже готово, вам осталось только реализовать алгоритм дешифровки. Суть его заключается в следующем: вам приходит целое неотрицательное число N , ваша задача – получить такое целое положительное число K , что произведение цифр числа K в точности равно числу N .

Входные данные.

В единственной строке дано число N ($0 \leq N \leq 10^{18}$).

Выходные данные.

В единственной строке выведите число K , произведение цифр которого в точности равно исходному числу N . Если ответов может быть несколько, то следует вывести наибольший возможный из них или слово «Infinity» (без кавычек), если результат больше, чем 2^{50} . Если такое число подобрать нельзя, выведите «No solution» (без кавычек).

Примеры.

Входные данные	Выходные данные
26	No solution
2251799813685248	Infinity

Задача Е. Есть время на отдых

Time limit: 1 с.

Memory limit: 256 МВ

В свое свободное время учащиеся одного учебного военного центра любят развлекать себя задачами по математике и программированию. Узнав об этом, их командир решил дать им одну несложную задачу: имеются два слова, состоящие из строчных латинских букв, нужно определить, сколько нужно сделать циклических сдвигов первого слова вправо, чтобы получилось второе слово. Курсанты научились достаточно быстро решать эту задачу для слов небольшого размера, теперь же они хотят написать программу, которая будет решать задачу командира для достаточно длинных слов.

Входные данные.

Весь ввод состоит из двух строк: в первой строке находится первое слово, во второй — второе. Слова состоят только из строчных латинских букв, длина обоих слов одинаковая и не превышает 500000 знаков.

Выходные данные.

В единственной строке выведите число N , которое означает, сколько циклических сдвигов первого слова вправо нужно сделать, чтобы получить второе слово.

Циклический сдвиг вправо — это перемещение последней буквы слова в его начало. Если есть несколько вариантов ответа, выведите минимальный из них.

Если второе слово нельзя получить сколь угодно большим числом циклических сдвигов первого слова, выведите -1 .

Примеры.

Входные данные	Выходные данные
abcde deabc	2
abcde deacb	-1

Задача F. Факультетские друзья

Time limit: 2 с.

Memory limit: 256 МВ

Когда на факультет военного обучения набирают новых студентов, многие из них не знакомы друг с другом. Двое друзей, попавших в одну роту, решили проверить, работает ли на факультете теория шести рукопожатий, которую выдвинули психологи Милгрэм и Трэверс. Суть теории заключается в том, что любые два человека на Земле разделены не более чем пятью уровнями общих знакомых, то есть, если переходить только от одного друга к другому, мы сможем достичь любого человека, посетив не больше, чем пятерых других людей. Они собрали список всех друзей на факультете, теперь необходимо написать программу, которая сможет обработать данные.

Входные данные.

В первой строке заданы два целых числа N и M ($1 \leq N, M \leq 5000$) — количество человек на факультете и количество пар друзей. В следующих M строках содержатся пары друзей — два целых положительных числа U и V , обозначающие порядковые номера друзей в общем списке факультета ($1 \leq U, V \leq 5000$). Мы предполагаем, что отношение дружбы обладает свойством коммутативности, то есть, если U друг V , то и V друг U .

Выходные данные.

В единственной строке выведите слово «Yes», если теория шести рукопожатий для данной группы студентов работает, и «No» в противоположном случае (без кавычек).

Примеры.

Входные данные	Выходные данные
4 4 1 2 3 1 2 4 2 3	Yes