Задача A. Шахматный конь

Имя входного файла:	knight.in
Имя выходного файла:	knight.out
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Вася решил научиться играть в шахматы. Он нашел книгу с записями партий и внимательно их изучает. Может быть, когда-нибудь Вася станет великим шахматистом, но пока он еще учится в начальной школе, и ему нелегко дается шахматная нотация. Больше всего трудностей у Васи вызывают ходы шахматного коня. Он попросил вас написать программу, которая сможет сообщить Васе, на какие клетки можно пойти конем с заданной клетки.

Вы, наверное, тоже знаете, что конь в шахматах всегда перемещается либо на две клетки по горизонтали и на одну по вертикали, либо на одну по горизонтали и на две по вертикали. Вертикали обозначаются маленькими латинскими буквами от `a` до `h`, а горизонтали — цифрами от 1 до 8. Любая клетка на шахматной доске обозначается буквой соответствующей вертикали и цифрой соответствующей горизонтали, например, `c6` или `e2`.

Формат входного файла

Во входном файле записаны только два символа — координаты клетки, в которой стоит конь.

Формат выходного файла

В выходной файл выведите координаты клеток, на которые за один ход может попасть конь, находящийся на заданной клетке.

Пример

knight.in	**knight.out**
b7	d6 c5 d8 a5

Задача B. Даты

Имя входного файла:	`dates.in`
Имя выходного файла:	`dates.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

При обработке различной информации часто приходится оперировать с данными о датах. В этой задаче вам нужно составить программу, которая сможет вывести число, месяц и день недели, которые наступят через k дней начиная с первого сентября 2005 года. Можно напомнить, что это был четверг.

Формат входного файла

Входной файл содержит одно целое число k ($0 \le k \le 1000$) — количество дней, после первого сентября, через которое наступит (или наступила) интересующая дата.

Формат выходного файла

В выходной файл ваша программа должна записать, какие день недели, число и месяц наступят по прошествии заданного времени. Результаты проверяются автоматически, поэтому вам следует придерживаться формата, показанного в примере.

Примеры

dates.in	dates.out
`0`	`Thursday, 01.09`
`5`	`Tuesday, 06.09`

Примечание

По-английски дни недели называются так: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday.

Задача C. Редактор

Имя входного файла:	`editor.in`
Имя выходного файла:	`editor.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Немногие знают, что первые версии текстового редактора из предыдущей задачи были написаны в России программистом Колей. Для этого он несколько месяцев почти не выходил из подвала, где стояли только диван и компьютер.

Вот одна из проблем, с которыми столкнулся Коля в те времена. Во время работы пользователь набирает какой-то текст, а так же может его редактировать. При этом, даже если итоговый результат полностью помещается на экран, в процессе работы отдельные строки могут иметь слишком большую длину. Мы не будем просить вас повторить Колин подвиг и заново написать редактор. Определите, какой максимальной длины строка получалась в течение набора текста, если вам известно, какие клавиши и в каком порядке нажимал пользователь.

Формат входного файла

Во входном файле записана строка из различных символов — последовательность кнопок на клавиатуре, которые нажимал пользователь. Переводы строк заменены на символ ‘\’. Первые версии редактора поддерживали три управляющие команды, которые закодированы следующим образом: ‘<’ — удаление предыдущего символа (если курсор находится в начале строки, и эта строка не первая, то удаляется предшествующий перевод строки); ‘^’ — перемещение в конец предыдущей строки (игнорируется, если курсор находится на первой строке); ‘|’ — перемещение в конец следующей строки (игнорируется, если курсор находится на последней строке). Все остальные символы, содержащиеся в файле, имеют коды от 32 (пробел) и выше и должны пониматься как есть. Количество нажатий клавиш не превосходит 100000.

Формат выходного файла

В выходной файл выведите одно целое число — максимальную длину строки, которая была достигнута в течение работы редактора.

Примеры

editor.in

```
Hello, World???<<<!
```

editor.out

```
15
```

editor.in

```
Hello, World?\This is^<!!!| a sample.
```

editor.out

```
17
```

Задача D. Палиндром

Имя входного файла:	`palin.in`
Имя выходного файла:	`palin.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Одно из известных развлечений со словами - составление палиндромов. Палиндромом называется предложение, которое, после удаления из него всех пробелов и знаков препинания, читается одинаково справа налево и слева направо. Создатели одного известного текстового редактора пишут новую версию модуля для проверки орфографии. Они хотят реализовать возможность вывода подсказки для пользователя на тот случай, если он допустил опечатку при наборе какого-нибудь палиндрома. Конечно же, они решили обратиться именно к вам.

Более точно, по заданной строке нужно определить, может ли она быть результатом замены, удаления или добавления одного символа в некотором палиндроме. При этом строчные и прописные латинские буквы не различаются, а все остальные символы должны игнорироваться.

Формат входного файла

Во входном файле содержится заданная строка. Гарантируется, что она содержит хотя бы одну букву. Длина строки не превосходит 100000.

Формат выходного файла

В первой строке выходного файла выведите `YES`, если строка может быть получена каким-нибудь из описанных выше преобразований из некоторого палиндрома, и `NO` в противном случае.

В случае положительного ответа во второй строке выведите какой-нибудь из палиндромов, в которых мог допустить опечатку пользователь.

Примеры

palin.in
`Never odd or even`
palin.out
`YES` `NeVErODDoREVen`

palin.in
`Eat it!`
palin.out
`NO`

palin.in
`Mums are not set as a test on Erasmus.`
palin.out
`YES` `Sums are not set as a test on Erasmus.`

Задача E. Не показывайте учителям химии

Имя входного файла:	chem.in
Имя выходного файла:	chem.out
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Запись химической реакции всегда содержит описания нескольких веществ. В свою очередь, описание одного химического вещества — строка, в которой входящие в него атомы химических элементов перечисляются в определенном порядке. При этом последовательности из двух и более одинаковых атомов, идущих подряд, группируются: записывается сокращенное название химического элемента и количество одинаковых элементов подряд. Например, вместо `HH` пишут `H2`. Обозначения химических элементов состоят из одной или двух латинских букв, из которых первая — прописная, а вторая - строчная. В этой задаче не будут рассматриваться более сложные правила. Например, не используются скобки. Вы должны проверить, что заданная последовательность символов подходит под данное выше описание формулы химического вещества. При этом не нужно рассматривать корректность заданной строки, исходя из каких-либо других соображений, даже если они продиктованы здравым смыслом.

Формат входного файла

На единственной строке входного файла записана последовательность символов, содержащая только цифры и строчные и прописные латинские буквы. Гарантируется, что в последовательности перед каждой строчной буквой идет прописная, а все однобуквенные и двухбуквенные подстроки, начинающиеся с прописной буквы - правильные обозначения химических элементов (поэтому здесь даже не приводится их список). Длина последовательности не превосходит 1000 символов.

Формат выходного файла

В выходной файл выведите одно слово: `YES`, если данная строка подходит под упрощенное описание формулы химического вещества из условия и `NO`, если не подходит.

Примеры

chem.in	chem.out
OHNaOHNa	YES
H2O	YES
HH	NO
CHC	YES

Задача F. Треугольники

Имя входного файла:	`triangles.in`
Имя выходного файла:	`triangles.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Когда Вова зашел в класс математики, на школьной доске было в беспорядке нарисовано некоторое количество треугольников. Вове очень понравилась идея разрисовывания доски треугольниками, и он принялся изучать ситуацию более внимательно. Вова подумал, что наиболее простой и быстрый способ ровно нарисовать такое количество треугольников - вырезать из какого-нибудь материала шаблон и обводить его на доске мелом. Кроме того Вова предположил, что тот, кто рисовал треугольники, должно быть, очень спешил, и потому перемещал шаблон, не отрывая никакую его часть от доски.

Помогите Вове проверить его гипотезу, если он готов сообщить вам тщательно измеренные координаты вершин треугольников.

Формат входного файла

В первой строке входного файла записано одно целое число — количество треугольников n ($2 \le n \le 100$). Последующие n строк описывают треугольники, строка $i + 1$ содержит 3 пары положительных целых чисел, не превосходящих 10000 — координаты вершин треугольника номер i.

Формат выходного файла

В выходной файл выведите ответ на задачу: `YES`, если все треугольники можно было нарисовать, обводя один шаблон и не отрывая этот шаблон от доски, и `NO` в противном случае.

Примеры

triangles.in	**triangles.out**
3 1 1 1 2 2 1 2 1 2 2 1 2 1 1 2 1 2 2	YES
2 1 1 4 1 2 2 1 1 4 1 3 2	NO

Задача G. Единицы

Имя входного файла:	`ones.in`
Имя выходного файла:	`ones.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

На уроках информатики вас, наверное, учили переводить числа из одних систем счисления в другие и выполнять другие подобные операции. Пришло время продемонстрировать эти знания. Найдите количество единиц в двоичной записи заданного числа.

Формат входного файла

Во входном файле содержится единственное целое число n $(0 \le n \le 2000000000)$.

Формат выходного файла

Выходной файл должен содержать одно число — количество двоичных единиц в записи числа n.

Примеры

ones.in	**ones.out**
5	2
7	3

Задача H. Гадание

Имя входного файла:	`fortune.in`
Имя выходного файла:	`fortune.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Как и многие другие девочки, Маша любит разные гадания. Некоторое время назад Маша узнала новый способ гадать на числах — для какого-нибудь интересующего ее натурального числа n надо посчитать сумму всех чисел, на которые n делится без остатка. Маша не очень любит арифметику, и попросила вас написать программу, которая автоматизирует процесс гадания.

Формат входного файла

Во входном файле содержится натуральное число n ($1 \leq n \leq 1000$), которое Маша была вынуждена сообщить.

Формат выходного файла

В выходной файл выведите сумму всех натуральных делителей числа n.

Примеры

fortune.in	**fortune.out**
6	12
10	18