

Задача А. Выравнивание кода

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

Вы работаете в команде, которая пишет Невероятно Настраиваемый Редактор Кода, который фактически является текстовым редактором с некоторыми фишками. Вы работаете над модулем, который берет кусок кода, содержащий некоторые выражения или другую информацию, и выравнивает каждую колонку на фиксированную вертикальную позицию, но при этом результирующий код остается настолько коротким, насколько это возможно. То есть, все первые слова в каждой линии находятся на позиции $p_1 = 1$; вторые слова в каждой линии находятся на минимально возможной позиции p_2 , так что все концы первых слов находятся на или перед позицией $p_2 - 2$; третьи слова в каждой линии находятся на минимально возможной позиции p_3 , так что все концы вторых слов находятся на или перед позицией $p_3 - 2$, и так далее.

Код состоит из нескольких линий. Каждая линия содержит одно или несколько слов разделенных пробелами. Каждое слово содержит маленькие или большие латинские буквы и символы ASCII: пунктуационные знаки, разделители и другие непробельные символы. (ASCII коды с 33 до 126 включительно). Пробел имеет ASCII код 32.

Формат входного файла

Входной файл содержит одну или несколько линий кода. Все линии (включая последнюю) заканчиваются стандартным символом конца строки. Каждая линия содержит по крайней мере одно слово, каждое слово имеет длину от 1 до 80 символов включительно. Слова разделены одним или несколькими пробелами. Каждая линия во входном файле не превышает 180 символов. Всего в файле не более 1000 линий.

Формат выходного файла

Выведите в выходной файл отформатированный, выравненный код, который содержит такое же число линий, с такими же словами, слова идут в том же порядке, без пробелов в начале или конце линии. Слова должны быть отделены одним или несколькими пробелами, так что i -ое слово в каждой линии начинается на позиции p_i .

Пример

Символ ‘`␣`’ в примере ниже обозначает пробел (ASCII код 32).

<code>alignment.in</code>	<code>alignment.out</code>
<code>␣␣start:␣␣integer;␣␣␣␣//␣␣begins␣here</code> <code>stop:␣integer;␣␣//␣␣ends␣here␣␣</code> <code>␣s:␣␣string;␣␣␣</code> <code>c:␣␣␣char;␣␣//␣␣temp␣</code>	<code>start:␣integer;␣␣//␣␣begins␣here</code> <code>stop:␣␣integer;␣␣//␣␣ends␣␣␣here</code> <code>s:␣␣␣␣␣string;</code> <code>c:␣␣␣␣␣char;␣␣␣␣//␣␣temp</code>

Задача В. Логи Apache

Имя входного файла:	<code>apache.in</code>
Имя выходного файла:	<code>apache.out</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

21 декабря в Центр была передана информация от Штирлица — лог сервера Apache, отражающий доступ к ключевому интернет-ресурсу, расположенному в государстве Borderland. В результате успешной спецоперации в Borderland сложилась тупиковая ситуация вокруг референдума о том, какая из букв нового алфавита должна быть первой. После предыдущих итераций выбор сузился до **Y** и **Z**. Очередная итерация процесса выхода из кризиса намечалась уже через пять дней. Для того чтобы восстановить последовательность запросов, сотрудники Центра, не совсем пришедшие в себя после Дня Чекиста, решили перевести события лога к временной зоне секретной базы, на которой разрабатывается новая диверсия.

Формат входного файла

В первой строке записана временная зона секретной базы, к которой нужно преобразовать лог. Начиная со второй строки, задан сам лог. Строки лога Apache всегда имеют вид

$$\langle ip \rangle - \langle username \rangle \langle date \rangle \langle extra\ fields \rangle$$

Поле *ip* представляет собой ip-адрес узла, сделавшего запрос, в стандартном формате. Второе поле всегда содержит один символ '-'. Имя пользователя состоит из произвольного ненулевого количества латинских букв, символов '-' и кавычек. Формат даты следующий:

$$[\langle \text{день} \rangle / \langle \text{месяц} \rangle / \langle \text{год} \rangle : \langle \text{часы} \rangle : \langle \text{минуты} \rangle : \langle \text{секунды} \rangle \langle \text{временная зона} \rangle]$$

Временная зона всегда записывается ровно пятью символами:

$$\langle \text{знак} \rangle \langle \text{часы} \rangle \langle \text{минуты} \rangle$$

Знак принимает значение '+', если часы в данной временной зоне опережают часы в зоне GMT (Greenwich Mean Time), в противном случае ставится знак '-'.

День, часы, минуты и секунды всегда состоят ровно из двух цифр (как и в поле даты, так и во всех записях временных зон), год — из четырех. Месяц записан сокращением на английском языке и может принимать одно из следующих значений: **Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec**.

Допустимые во входном файле номера года лежат в диапазоне от 1980 до 2099. Високосными считаются года, номер которых делится на 4.

Дополнительные поля состоят из произвольного количества символов с кодами от 32 до 126.

В записях временной зоны поле часов всегда содержит значение, не превосходящее 12. Размер входного файла не превосходит одного мегабайта. Все даты корректны и проверка их не требуется.

Формат выходного файла

Результатом работы программы является лог Apache, в котором все времена и даты преобразованы к заданной временной зоне. Все остальные поля в каждой строке лога должны оставаться без изменений. Имейте в виду, что проверка производится автоматически, поэтому все даты и времена должны содержать по две цифры в полях дней, часов, минут и секунд, три символа в поле месяца (case-sensitive), и четыре цифры в поле года.

Сборы школьников Санкт-Петербурга
14 марта, группа А

Пример

apache.in
-0200 195.19.224.104 - abc [02/Dec/2004:18:25:19 +0100] "GET / HTTP/1.1" 304 - 195.19.224.104 - - [02/Dec/2004:18:25:19 +0100] "GET /main.css HTTP/1.1" 304 - 195.19.224.79 - - [02/Dec/2004:20:58:20 +0300] "GET /tts HTTP/1.1" 302 293 195.19.224.79 - - [02/Dec/2004:20:58:20 +0300] "GET /tts/ HTTP/1.1" 302 303 207.46.98.42 - - [03/Dec/2004:04:39:32 +0300] "GET /robots.txt HTTP/1.0"
apache.out
195.19.224.104 - abc [02/Dec/2004:15:25:19 -0200] "GET / HTTP/1.1" 304 - 195.19.224.104 - - [02/Dec/2004:15:25:19 -0200] "GET /main.css HTTP/1.1" 304 - 195.19.224.79 - - [02/Dec/2004:15:58:20 -0200] "GET /tts HTTP/1.1" 302 293 195.19.224.79 - - [02/Dec/2004:15:58:20 -0200] "GET /tts/ HTTP/1.1" 302 303 207.46.98.42 - - [02/Dec/2004:23:39:32 -0200] "GET /robots.txt HTTP/1.0"

Задача D. Раскраска куба

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

В далекой галактике $\mathbb{X}\ddot{o}\mathbb{S}$ живет робот по имени А. С. Магнефайзер. Мистер Магнефайзер — или просто “А. С. М.,” как он предпочитает, чтобы его называли — предельно счастлив, когда он раскрашивает красивые фигуры.

Вчера старый художник пришел к мистеру Магнефайзеру и предложил ему работу. Художнику необходимо раскрасить куб, который является важной частью его следующей выставки. Но только у него есть свое *видение*, как куб должен выглядеть. *Видение грани* — это то, как куб должен выглядеть, когда смотришь на определенную грань. Оно включает себя цвет грани, на которое смотрит наблюдатель (“главная” грань) и список цветов тех граней, которые имеют общее ребро с главной. Порядок прилегающих граней не фиксирован, потому что «Видение грани» это впечатление, а не мелкие детали.

Видение куба — это «Видение грани» всех граней куба, но, опять таки, без какого-либо порядка.

Хотя у мистера Магнефайзера большой опыт в раскрашивании кубов, так же как и других фигур, но это задача кажется ему слишком сложной. Помогите ему! Вам дано «Видение куба», найдите способ раскрасить его.

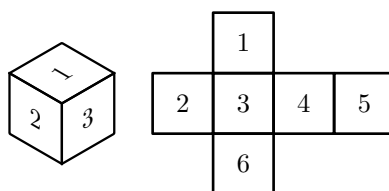
Формат входного файла

В первой строчке входного файла содержится 6 слов. Каждое слово описывает «Видение грани» каждой грани куба. Описание содержит 5 слов, состоящих из английских заглавных букв — цвет главной грани (первая буква) и цвета прилегающих граней. Одинаковые буквы означают одинаковые цвета, а различные буквы — различные цвета.

Формат выходного файла

Если невозможно раскрасить куб, из-за того что некоторые «Видения грани» противоречат друг другу выведите “Impossible”.

Если есть единственный способ раскрасить куб (если раскраски могут получены друг из друга поворотом куба, то они одинаковы), выведите цвета граней в одной строчке от первой грани до шестой, как показано на рисунке ниже. Куб может быть повернут как угодно.



Если есть несколько способов раскрасить куб, выведите любые два из них в формате, показанном выше, каждый на отдельной строчке.

Примеры

<code>cube.in</code>	<code>cube.out</code>
ABCDE FGHIJ KLMNO PQRST UVWXY ZABCD	Impossible
OZZZZ ZOZZZ ZZOZZ ZZZOZ ZZZZO ZZZZZ	ZOZZZZ
ABCOO BCAOO CABOO OOOAB OOOBC OOOCA	ABC000 ACB000