

Задание 7: Строки

Для всех задач:

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда на тест
Ограничение по памяти: 64 МБ

Задача 1. Игра в алфавит

Два товарища играют в слова. Игроки по очереди приводят примеры слов из некоторой предметной области. Начинает Игрок 1. На каждом ходу игрок должен привести пример, который содержит указанную букву алфавита, которая изменяется на следующем шаге. В частности, Игрок 1 должен сначала привести пример слова, содержащего букву **a**. Игрок 2 должен затем привести пример слова, содержащего букву **b**. Игрок 1 должен на следующем шаге привести пример слова, содержащего **c**, и так далее. Если игрок не может назвать соответствующее слово, то он проигрывает, а другой игрок, соответственно, выигрывает.

Для заданной последовательности слов, которые по очереди произносили игроки, вам нужно написать программу, которая бы определяла, кто победил. Проигравшим считается тот игрок, который первым не смог привести соответствующий пример слова.

Входные данные

Во входном файле записана непустая строка, содержащая последовательность слов, разделенных одним пробелом. Все слова содержат маленькие латинские буквы. Количество слов не превышает 26. Длина одного слова может изменяться от 1 до 20.

Выходные данные

В выходной файл необходимо вывести одну из трёх возможных комбинаций слов. Если выигрывает первый игрок, то в выходной файл нужно вывести **PLAYER 1**. Если же выигрывает второй игрок, нужно вывести **PLAYER 2**. Если никто не выигрывает, то вывести **NO WINNER**.

Примеры

<i>input.txt</i>	<i>output.txt</i>
alpha beta gamma delta epsilon	PLAYER 2
january february march april may	PLAYER 1
cab cab cab ford chevy ford dodge honda	NO WINNER

Задача 2. Смешивание цветов

Известно, что можно смешивать различные цвета красок, чтобы получать другие цвета. В частности, для того чтобы получить зеленый цвет, нужно смешать желтый и синий, оранжевый может получиться смешиванием красного и желтого, а фиолетовый — из красного и синего. В магазине есть автоматическая линия для смешивания красок. В нее заряжены баллончики с красками разных цветов. Она может смешивать любые два баллончика. Ваша задача по заданным требуемым цветам (**green**, **orange** или **purple**) определить, какие баллончики смешивать.

Входные данные

Во входном файле записана строка, состоящая из двух слов, разделенных одним пробелом. Первое слово, записанное маленькими буквами, определяет требуемый цвет, один из трех: **green**, **orange** или **purple**. Второе слово состоит из последовательности больших латинских букв, каждая буква представляет какой-то цвет. Цвета красный, желтый и синий будут представлены буквами **R**, **Y** и **B**, соответственно. Другие буквы тоже могут в этом слове встречаться. Буквы **R**, **Y** и **B** встречаются во втором слове только один раз. Его длина может изменяться от 3 до 26.

Задание 7: Строки

Выходные данные

В выходной файл необходимо вывести два целых числа, разделенных пробелом. Эти числа задают номера баллончиков с краской, которые нужно смешать, чтобы получить нужный цвет. Целые нужно выводить в возрастающем порядке. Нумерация баллончиков начинается с 1.

Примеры

<i>input.txt</i>	<i>output.txt</i>
purple KWBYNXRS	3 7
green BARCYZ	1 5
orange GSWXBRY	6 7

Задача 3. Разность множеств

Дано два множества символов S_1 и S_2 . Вам нужно вычислить их разность $S_1 \setminus S_2$ — новое множество, в котором присутствуют только те элементы S_1 , которых нет в S_2 .

Входные данные

В единственной строке входного файла через пробел записано две непустые последовательности латинских букв в верхнем регистре. Эти последовательности задают два множества — S_1 и S_2 , соответственно. Каждый элемент множества встречается в последовательности не более одного раза. Количество элементов в каждом множестве не более 26.

Выходные данные

В выходной файл необходимо вывести последовательность символов, которые будут представлять разность заданных множеств. Элементы искомого множества выдавать в возрастающем порядке. Если результирующее множество — пусто, то вывести **Empty Set**.

Примеры

<i>input.txt</i>	<i>output.txt</i>
AEIOU AEIOUY	Empty Set
PQRSTUV UVWXY	PQRST
TEXAS COMPUTER	ASX

Задача 4. Слова

Перечислить все слова из заданной строки, не содержащие букв, из которых состоит первое слово.

Входные данные

Во входном файле записана строка, состоящая из слов, разделенных одним пробелом. Каждое слово состоит из маленьких латинских букв.

Выходные данные

В выходной файл необходимо вывести все слова, по одному на строку, обладающие заданным свойством, в порядке их встречаемости во входном файле.

Примеры

<i>input.txt</i>	<i>output.txt</i>
to be or not to be that is the question	be be is

Задание 7: Строки

Задача 5. Топики

Одним из простых методов определения темы статьи в газете является выявление наиболее часто встречающихся слов в ней. Естественно, маленькие служебные слова, такие как "the", "a" или "of" нужно игнорировать, т.к. они не влияют на содержание статьи.

Ваша задача — по заданным словам, которые нужно игнорировать, найти в предложенной статье наиболее часто встречающееся слово.

Входные данные

В первой строке входного файла задан список слов, которые нужно игнорировать. Между словами стоит ровно один пробел. Их количество не превышает 20.

Остальные строки входного файла содержат текст, в котором нужно найти наиболее встречающееся слово. Слова не разрываются концом строки. Каждая строка начинается с нового слова. Статья заканчивается двумя звездочками **. Регистр букв в словах нужно игнорировать. Однокоренные слова с разными окончаниями считаются разными словами. Знаки препинания тоже нужно игнорировать.

Выходные данные

В выходной файл необходимо вывести наиболее часто встречающееся в заданной статье слово. Это слово нужно вывести в верхнем регистре.

Примеры

<i>input.txt</i>	<i>output.txt</i>
the at in on to of is that The state finals of the Texas Computer Education Association Computer Programming Contest is today. Teams from all over the state of Texas are participating in the event. In last year's contest, each team brought their own computer.**	COMPUTER