

## Разделение выражения на лексемы

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

Задано числовое выражение, заканчивающееся точкой. Необходимо разбить его на лексемы и вывести каждую на новой строке. Гарантируется, что исходное выражение корректно.

В выражении могут встречаться знаки сложения, вычитания, умножения и скобки. Приоритет операций стандартный. Все числа в выражении целые и принадлежат диапазону `long`.

### Формат входных данных

Первая строка входного файла содержит заданное выражение. Его длина не превосходит 100 знаков. Гарантируется, что выражение заканчивается точкой.

### Формат выходных данных

Выведите все встречающиеся лексемы выражения по порядку и ровно по одной на каждой строке.

### Пример

<code>lexem.in</code>	<code>lexem.out</code>
<code>1+(2*2-3).</code>	<code>1</code> <code>+</code> <code>(</code> <code>2</code> <code>*</code> <code>2</code> <code>-</code> <code>3</code> <code>)</code>

## Значение выражения

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

Задано числовое выражение, заканчивающееся точкой. Необходимо посчитать его значение или сказать, что оно содержит ошибку. В выражении могут встречаться знаки сложения, вычитания, умножения и скобки. Приоритет операций стандартный. Все числа в выражении целые и принадлежат диапазону `long`. Также гарантируется, что все промежуточные вычисления уместятся в этот тип.

### Формат входных данных

Первая строка входного файла содержит заданное выражение. Его длина не превосходит 100 знаков. Гарантируется, что выражение заканчивается точкой.

### Формат выходных данных

Выведите в выходной файл значение этого выражения или слово «`WRONG`», если значение не определено.

### Примеры

<code>expr.in</code>	<code>expr.out</code>
<code>1+(2*2-3).</code>	<code>2</code>
<code>1+a+1.</code>	<code>WRONG</code>

## Новогоднее выражение

Имя входного файла: `expr2.in`  
Имя выходного файла: `expr2.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Задано новогоднее выражение, заканчивающееся точкой. Необходимо посчитать его значение или сказать, что оно содержит ошибку. В новогоднем выражении могут встречаться знаки сложения, вычитания, умножения и скобки, константы `Ded Moroz`, `Moroz` и `Snegurochka`, а также вызов функции `Podarok`, которая принимает одно число на вход и возвращает его, увеличенное на 5, если оно было положительно, или возвращает его модуль, если оно было меньше либо равно 0. Приоритет операций стандартный. Все числа в выражении целые и принадлежат диапазону `long`. Также гарантируется, что все промежуточные вычисления уместятся в этот тип.

### Формат входных данных

Первая строка входного файла содержит заданное выражение. Его длина не превосходит 200 знаков. Гарантируется, что выражение заканчивается точкой.

Значения констант:

<code>Ded Moroz</code>	2016
<code>Moroz</code>	-30
<code>Snegurochka</code>	10

### Формат выходных данных

Выведите в выходной файл значение этого новогоднего выражения или слово `WRONG`, если значение не определено.

### Примеры

<code>expr2.in</code>	<code>expr2.out</code>
<code>Podarok(Moroz-Ded Moroz)*2.</code>	4092
<code>Snegurochka-30.</code>	-20