

1. Дана строка, в которой возможны нарушения правил набора текстов. Рассматривается только проблема правильной расстановки пробелов вокруг запятой.

TOT, _KTO	адекватный и вменяемый человек
TOT, KTO	есть странности (или кончились знаки в твиттере)
TOT, _KTO	наглухо перекрытый олигофрен
TOT, _ _KTO	опасен для общества и самого себя, вызывайте скорую

Написать программу, оценивающую уровень автора текста.

Примечание 1. Для простоты считаем, что в строке может быть только один тип ошибки.

Примечание 2. Рисунок взят из Интернета. В своей программе не обязательно следовать прямым определениям автора текстов (см. второй столбик). Программа должна дополнительно выдавать верный текст.

Примечание 3. Дополнительно можно рассмотреть случаи «совсем большого разума»: предложение начинается (заканчивается) с запятой. Этот случай предложен студентами 2 группы (2024/2025 учебный год).

2. Дан текст, содержащий названия числительных от 0 до 20, записанные в именительном падеже и единственном числе. Обработать текст так, чтобы вместо числительных стояли соответствующие числовые значения. Например, для текста

Жили три поросенка, был у них один дом. Результат: Жили 3 поросенка, был у них 1 дом.

Указание: словари не использовать. Возможно, поможет список с именами числительных.

Усложненный вариант: названия числительных от 0 до 99

3. Дана строка, состоящая из нескольких слов. Вывести на печать порядковый номер начала каждого слова и длину слова.

Подготовка к тестированию по строкам

№ 1. Строка `s` инициализирована следующим образом: `s = 'rostov'`. Запишите результат работы каждого оператора (т.е., что будет присвоено переменной, стоящей в левой части оператора присваивания).

Указание. Результаты записывайте, не используя компьютер. В конце проверьте ответы, в которых сомневаетесь, с помощью Python.

<code>s[1] = 'R'</code>	
<code>s = s + 2025</code>	
<code>s = s[-1]</code>	
<code>k = len(s)</code>	
<code>s = 'R' + s[2:]</code>	
<code>s = s[::-1]</code>	
<code>k = s.count('o')</code>	
<code>p = s.index('o')</code>	
<code>p = s.index('O')</code>	
<code>s = ('3' * 10).count('33')</code>	
<code>s = '*'.join(['1', '2', '3']).split('-')</code>	

№ 2. Запишите результат работы оператора печати.

Указание. Результаты записывайте, не используя компьютер. В конце проверьте ответы, в которых сомневаетесь, с помощью Python.

<pre>s = 'Boltik' print(min(s) + '! ' + max(s))</pre>	
<pre>s = 'скоро лето' print(s[::-2])</pre>	
<pre>s = 'summer 2025' print(s[1::2])</pre>	
<pre>s = 'summer' print(2*(s[1] + s[len(s)//2] + s[-1]))</pre>	
<pre>s = 'mmcs. 1 course' print(s.replace('s', 'S', 1))</pre>	
<pre>Date = '13 March, 2025 - 20 March, 2025' m = Date[Date.find('')+1:Date.find(',')] print(m)</pre>	
<pre>def f(s): s1 = s[:s.find(' ')] s2 = s[s.rfind('')+1:] s0 = s[s.find(''):s.rfind('')+1] return s2 + s0 + s1 s = 'весна 13 марта теплый день' print(f(s))</pre>	
<pre>def f(s, s1, s2): D = len(s1) while s.find(s1) != -1: k = s.find(s1) s = s[:k] + s2 + s[k+D:] return s s = 'Dog-dog' s = f(f'{s}. {s}', 'Dog', 'Cat') s = f(s, 'dog', 'kitten') print(s)</pre>	
<pre>def sort_(sx): return sx[-1] a = ['56', '55', '9', '123', '1'] a.sort(key=sort_) print(a)</pre>	

3А. Требуется заменить во введенной строке s все слова winter на spring. Выберите верный оператор присваивания

- 1) `s = s.replace('winter', 'spring ', s.count('winter'))`
- 2) `s = s.replace('spring ', 'winter', s.count('winter'))`
- 3) `s = s.replace('winter', 'spring ', 1)`
- 4) `s = s.replace('spring ', 'winter')`

3Б. Требуется инвертировать часть строки между первым и последним символами. Выберите срез строки `s`, который позволит это сделать.

1) `s = s[len(s)-1:-len(s):-1]`

2) `s = s[len(s)-2:-len(s):-1]`

3) `s = s[-1:-len(s):1]`

4) `s = s[len(s):1:-1]`

4А. Дана строка `s`. В следующем фрагменте кода вычисляется значение переменной `k`:

```
k = 0
```

```
for t in s:
```

```
    if t == '-':
```

```
        k += 1
```

Приведите пример использования метода строк, заменяющего все четыре строки кода.

4Б. Даны три строки `s`, `s1`, `s2`. В следующем фрагменте кода изменяется строка `s`:

```
d = len(s1)
```

```
if s.find(s1) != -1:
```

```
    k = s.find(s1)
```

```
    s = s[:k] + s2 + s[k+d:]
```

```
print(s)
```

Приведите пример использования метода строк, заменяющего все пять строк кода.
