

9.4 Методы строк. Часть 2

Шаг 1

Тема урока: строки

1. Методы `count()`
2. Метод `startswith()`
3. Метод `endswith()`
4. Метод `find()`
5. Метод `rfind()`
6. Метод `index()`
7. Метод `rindex()`
8. Метод `strip()`
9. Метод `lstrip()`
10. Метод `rstrip()`
11. Метод `replace()`

Аннотация. Строковый тип данных, основные методы поиска и замены.

Поиск и замена

Методы поиска и замены строк внутри других строк.

Каждый метод в этой группе поддерживает необязательные аргументы `<start>` и `<end>`. Как и в строковых срезах, действие метода ограничено частью исходной строки, начинающейся с позиции символа `<start>` и продолжающейся вплоть до позиции символа `<end>`, но не включающей её. Если параметр `<start>` указан, а параметр `<end>` нет, то метод применяется к части исходной строки от `<start>` до конца строки. Если параметры не заданы, то подразумевается, что `<start> = 0`, `<end> = len(s)`.

Метод `count()`

Метод `count(<sub>, <start>, <end>)` считает количество **непересекающихся** вхождений подстроки `<sub>` в исходную строку `s`.

Приведённый ниже код:

```
s = 'foo goo moo'
print(s.count('oo'))
print(s.count('oo', 0, 8)) # подсчет с 0 по 7 символов
```

выводит:

```
3
2
```

Метод `startswith()`

Метод `startswith(<suffix>, <start>, <end>)` определяет, **начинается** ли исходная строка `s` подстрокой `<suffix>`. Метод возвращает значение `True`, если исходная строка начинается с подстроки `<suffix>`, или `False` в противном случае.

Приведённый ниже код:

```
s = 'foobar'  
print(s.startswith('foo'))  
print(s.startswith('baz'))
```

выводит:

```
True  
False
```

Метод endswith()

Метод `endswith(<suffix>, <start>, <end>)` определяет, **оканчивается ли** исходная строка `s` подстрокой `<suffix>`. Метод возвращает значение `True`, если исходная строка оканчивается на подстроку `<suffix>`, или `False` в противном случае.

Приведённый ниже код:

```
s = 'foobar'  
print(s.endswith('bar'))  
print(s.endswith('baz'))
```

выводит:

```
True  
False
```

Методы find(), rfind()

Метод `find(<sub>, <start>, <end>)` находит **индекс первого вхождения** подстроки `<sub>` в исходной строке `s`. Если строка `s` не содержит подстроки `<sub>`, то метод возвращает значение `-1`. Мы можем использовать данный метод наравне с оператором `in` для проверки: содержит ли заданная строка некоторую подстроку или нет.

Приведённый ниже код:

```
s = 'foo bar foo baz foo qux'  
print(s.find('foo'))  
print(s.find('bar'))  
print(s.find('qu'))  
print(s.find('python'))
```

выводит:

```
0  
4  
20  
-1
```

Метод `rfind(<sub>, <start>, <end>)` идентичен методу `find(<sub>, <start>, <end>)`, за тем исключением, что он ищет первое вхождение подстроки `<sub>`, начиная с конца строки `s`.

Методы index(), rindex()

Метод `index(<sub>, <start>, <end>)` идентичен методу `find(<sub>, <start>, <end>)`, за тем исключением, что он **вызывает ошибку** `ValueError: substring not found` во время выполнения программы, если подстрока `<sub>` не найдена.

Метод `rindex(<sub>, <start>, <end>)` идентичен методу `index(<sub>, <start>, <end>)`, за тем исключением, что он ищет первое вхождение подстроки `<sub>`, начиная с конца строки `s`.



Методы `find()` и `rfind()` являются более безопасными, чем `index()` и `rindex()`, так как не приводят к возникновению ошибки во время выполнения программы.

Метод strip()

Метод `strip()` возвращает копию строки `s`, у которой удалены все пробелы, стоящие **в начале и конце** строки.

Приведённый ниже код:

```
s = '    foo bar foo baz foo qux    '
print(s.strip())
```

выводит:

```
foo bar foo baz foo qux
```

Метод lstrip()

Метод `lstrip()` возвращает копию строки `s`, у которой удалены все пробелы, стоящие **в начале** строки.

Приведённый ниже код:

```
s = '    foo bar foo baz foo qux    '
print(s.lstrip())
```

выводит (символом `_` обозначены пробелы):

```
foo bar foo baz foo qux_____
```

Метод rstrip()

Метод `rstrip()` возвращает копию строки `s`, у которой удалены все пробелы, стоящие **в конце** строки.

Приведённый ниже код:

```
s = '    foo bar foo baz foo qux    '
print(s.rstrip())
```

выводит (символом `_` обозначены пробелы):

```
_____foo bar foo baz foo qux
```

Метод replace()

Метод `replace(<old>, <new>)` возвращает копию `s` **со всеми** вхождениями подстроки `<old>`, заменёнными на `<new>`.

Приведённый ниже код:

```
s = 'foo bar foo baz foo qux'
print(s.replace('foo', 'grault'))
```

выводит:

```
grault bar grault baz grault qux
```

Метод `replace()` может принимать необязательный третий аргумент `<count>`, который определяет количество замен.

Приведённый ниже код:

```
s = 'foo bar foo baz foo qux'
print(s.replace('foo', 'grault', 2))
```

выводит:

```
grault bar grault baz foo qux
```

Примечания

Примечание. Методы `strip()`, `lstrip()`, `rstrip()` могут принимать на вход необязательный аргумент `<chars>`. Необязательный аргумент `<chars>` – это строка, которая определяет набор символов для удаления.

Приведённый ниже код:

```
s = '-++-+abc+-+-'  
  
print(s.strip('+-'))  
print(s.rstrip('+-'))  
print(s.lstrip('+-'))
```

выводит:

```
abc  
-++-+abc  
abc+-+-
```

❤️ Happy Pythoning! 🧑

Шаг 2

Что покажет приведённый ниже код?

```
s = 'aabbaAccDDaa'  
s = s.lower()  
print(s.count('a'))
```

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/2>

Шаг 3

Что покажет приведённый ниже код?

```
s = 'www.stepik.org'  
print(s.startswith('www'))
```

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/3>

Шаг 4

Что покажет приведённый ниже код?

```
s = 'www.stepik.org'  
print(s.endswith('.ru'))
```

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/4>

Шаг 5

Что покажет приведённый ниже код?

```
s = 'I learn Python language. Python - awesome!'
print(s.find('Python'))
```

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/5>

Шаг 6

Что покажет приведённый ниже код?

```
s = '      I learn Python language
print(s.strip())
```

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/6>

Шаг 7

Что покажет приведённый ниже код?

```
s = 'abcdababa'
print(s.replace('ab', '123'))
```

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/7>

Шаг 8

Количество слов

На вход программе подаётся строка текста, состоящая из слов, разделённых ровно одним пробелом. Напишите программу, которая подсчитывает количество слов в ней.

Формат входных данных

На вход программе подаётся строка текста.

Формат выходных данных

Программа должна вывести количество слов.

Примечание. Стока текста не содержит пробелов в начале и конце.

► Тестовые данные

Sample Input 1:

Hello world

Sample Output 1:

2

Sample Input 2:

Python

Sample Output 2:

1

Sample Input 3:

In 2010, someone paid 10k Bitcoin for two pizzas.

Sample Output 3:

9

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/8>

Шаг 9

Минутка генетики

На вход программе подаётся строка генетического кода, состоящая из букв А (аденин), Г (гуанин), Ц (цитозин) и Т (тимин). Напишите программу, которая подсчитывает сколько аденина, гуанина, цитозина и тимина входит в данную строку генетического кода.

Формат входных данных

На вход программе подаётся строка генетического кода, состоящая из символов А, Г, Ц, Т, а, г, ц, т.

Формат выходных данных

Программа должна вывести, сколько гуанина, тимина, цитозина, аденина входит в данную строку генетического кода.

Примечание. Стока не содержит других символов, кроме А, Г, Ц, Т, а, г, ц, т.

► Тестовые данные

Sample Input 1:

АаагГГЦЦцТТtttT

Sample Output 1:

Аденин: 3
Гуанин: 2
Цитозин: 3
Тимин: 5

Sample Input 2:

ааггццттААГГЦЦтт

Sample Output 2:

Аденин: 4
Гуанин: 4
Цитозин: 4
Тимин: 4

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/9>

Шаг 10

Очень странные дела

Джим Хоппер с помощью радиоприёмника пытается получить сообщение Оди. На приёмник ему поступает n различных последовательностей кода Морзе. Декодировав их, он получает последовательности из цифр и букв строчного латинского алфавита. При этом **только в сообщениях Оди** содержится число 11, причём минимум 3 раза. Помогите определить Джиму количество сообщений от Оди.

Формат входных данных

В первой строке подаётся число n – количество сообщений, в последующих n строках вводятся сами сообщения.

Формат выходных данных

Программа должна вывести количество сообщений от Оди.

Примечание. Обратите внимание, что в сообщениях Оди вхождения числа 11 должны быть **непересекающимися**. Другими словами, если мы нашли вхождение числа 11, то следующее вхождение должно начинаться строго после окончания предыдущего. Например, в строке `'111'` содержится одна такая последовательность, в то время как в `'1111'` их уже две.

► Тестовые данные 

Sample Input 1:

```
3
11helpme11jim11
avengers141414atta11ck
k1lg0re11111l
```

Sample Output 1:

```
1
```

Sample Input 2:

```
5
eowi11fjoigei23do11i23gf2983fj19f1120f92f23
i2oj2fjoerfjpwmkewfewkekef
wewewewew111wweeweweewew11w
111111
krewjrgeorgjogergiurg
```

Sample Output 2:

```
2
```

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/10>

Шаг 11

Количество цифр

На вход программе подаётся строка текста. Напишите программу, которая подсчитывает количество цифр в данной строке.

Формат входных данных

На вход программе подаётся строка текста.

Формат выходных данных

Программа должна вывести количество цифр в данной строке.

► Тестовые данные 

Sample Input 1:

nezabud dl-6

Sample Output 1:

1

Sample Input 2:

l33t 3301

Sample Output 2:

6

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/11>

Шаг 12

.com or .ru 

На вход программе подаётся строка текста. Напишите программу, которая проверяет, что строка заканчивается подстрокой

.com или .ru .

Формат входных данных

На вход программе подаётся строка текста.

Формат выходных данных

Программа должна вывести «YES» (без кавычек), если введённая строка заканчивается подстрокой .com или .ru , или «NO» (без кавычек) в противном случае.

► Тестовые данные 

Sample Input 1:

www.stepik.org

Sample Output 1:

NO

Sample Input 2:

www.google.com

Sample Output 2:

YES

Sample Input 3:

www.yandex.ru

Sample Output 3:

YES

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/12>

Шаг 13

Самый частотный символ

На вход программе подаётся строка текста. Напишите программу, которая выводит на экран символ, который появляется наиболее часто.

Формат входных данных

На вход программе подаётся строка текста. Текст может содержать строчные и заглавные буквы английского и русского алфавита, а также цифры.

Формат выходных данных

Программа должна вывести символ, который появляется наиболее часто.

Примечание 1. Если таких символов несколько, следует вывести последний по порядку символ.

Примечание 2. Следует различать заглавные и строчные буквы, а также буквы русского и английского алфавита.

► Тестовые данные



Sample Input 1:

aaaabbc

Sample Output 1:

a

Sample Input 2:

abaabbcccc

Sample Output 2:

c

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/13>

Шаг 14

Первое и последнее вхождение

На вход программе подаётся строка текста. Если в этой строке буква «f» встречается только один раз, выведите её индекс. Если она встречается два и более раза, выведите индексы её первого и последнего вхождения на одной строке, разделённые символом пробела. Если буква «f» в данной строке не встречается, следует вывести «NO» (без кавычек).

Формат входных данных

На вход программе подаётся строка текста.

Формат выходных данных

Программа должна вывести текст в соответствии с условием задачи.

► Тестовые данные 

Sample Input 1:

abcdefg

Sample Output 1:

5

Sample Input 2:

abcdefgfhfabc

Sample Output 2:

5 9

Sample Input 3:

abcd

Sample Output 3:

NO

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/14>

Шаг 15

Удаление фрагмента

На вход программе подаётся строка текста, в которой буква «h» встречается минимум два раза. Напишите программу, которая удаляет из этой строки первое и последнее вхождение буквы «h», а также все символы, находящиеся между ними.

Формат входных данных

На вход программе подаётся строка текста.

Формат выходных данных

Программа должна вывести текст в соответствии с условием задачи.

► Тестовые данные 

Sample Input 1:

ahahahahaha

Sample Output 1:

aa

Sample Input 2:

hh

Sample Output 2:

Чтобы решить это задание откройте <https://stepik.org/lesson/303083/step/15>