Университет ИТМО

Факультет программной инженерии и компьютерной техники

Лабораторная работа №3

по информатике

«Регулярные выражения»

Вариант №57

Выполнил:

Степанов Артур

Группа: P3112

Приняла:

Рудникова Тамара Владимировна

???

?

г. Санкт-Петербург

2021 г.

Оглавление

[Основное задание 3](#_Toc85471959)

[Текст задания 3](#_Toc85471960)

[Исходный код программы 3](#_Toc85471961)

[Вывод программы 3](#_Toc85471962)

[Доп. Задание №1 3](#_Toc85471963)

[Текст задания 3](#_Toc85471964)

[Исходный код программы 4](#_Toc85471965)

[Вывод программы 4](#_Toc85471966)

[Доп. Задание №2 4](#_Toc85471967)

[Текст задания 4](#_Toc85471968)

[Исходный код программы 4](#_Toc85471969)

[Вывод программы 5](#_Toc85471970)

[Вывод 5](#_Toc85471971)

# Основное задание

## Текст задания

1) Реализуйте программный продукт на языке Python, используя регулярные выражения по варианту, представленному в таблице.

2) Для своей программы придумайте минимум 5 тестов. Каждый тест является отдельной сущностью, передаваемой регулярному выражению для обработки. Для каждого теста необходимо самостоятельно (без использования регулярных выражений) найти правильный ответ. После чего сравнить ответ, выданный программой, и полученный самостоятельно.

3) Программа должна считать количество смайликов определённого вида (вид смайлика описан в таблице вариантов) в предложенном тексте. Все смайлики имеют такую структуру: [глаза][нос][рот]. Вариантом является различные наборы глаз, носов и ртов.

## Исходный код программы

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | **import** **re**  tests = ["ri:-)pgujk:-(fuu:-(urifreoisd:-(kjfepi;uer:(",  ";oiuytgf:-(gbnm,.;'[pvfde56:-(7890-098765434",  "oiuyt:-(redsdxcvbhji:-(oiuyhgvb:-(hjklop;lkj",  "bkjkninl;lmojiph:-(pijknliug67568y0u9-i0:-(",  "uugiy8599870-09=09087546687986079\*%#:-(",  "pei4holkwlesdvoetkn;lwreh4865783470359-02i3["]  **for** testnumber **in** range(len(tests)):  str = tests[testnumber]  s = re.findall(r":-\(", str)  k=len(s)  print(k) |

## Вывод программы

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | 3  2  3  2  1  0 |

# Доп. Задание №1

## Текст задания

1) Реализуйте программный продукт на языке Python, используя регулярные выражения по варианту, представленному в таблице.

2) Для своей программы придумайте минимум 5 тестов. Каждый тест является отдельной сущностью, передаваемой регулярному выражению для обработки. Для каждого теста необходимо самостоятельно (без использования регулярных выражений) найти правильный ответ. После чего сравнить ответ, выданный программой, и полученный самостоятельно.

## Исходный код программы

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21 | **import** **re**  tests = ["Вечер за окном./Еще один день прожит./Жизнь скоротечна",  "Как вишня расцвела!/Она с коня согнала/И князя-гордеца.",  "сао лучшее аниме",  "покоя/на фоне безмятежных облаков/по одичалым веткам",  "Как полон сад дыханием цветов/Так мы полны азартом"]  **for** testnumber **in** range(len(tests)):  str = tests[testnumber]  l = re.split('/', str)  **if** len(l)==**3**:  a = []  **for** i **in** range(**3**):  s = re.findall(r"е|ы|а|о|э|я|и|ю|ё|у|Е|Ы|А|О|Э|Я|И|Ю|Ё|У", l[i])  k=len(s)  a.append(k)  **if** a[**0**]==**5** **and** a[**1**]==**7** **and** a[**2**]==**5**:  print("Хайку!")  **else**:  print("Не хайку.")  **else**:  print("Не хайку. Должно быть 3 строки.") |

## Вывод программы

|  |  |
| --- | --- |
| 1  2  3  4  5 | Хайку!  Не хайку.  Не хайку. Должно быть 3 строки.  Не хайку.  Не хайку. Должно быть 3 строки. |

# Доп. Задание №2

## Текст задания

1) Реализуйте программный продукт на языке Python, используя регулярные выражения по варианту, представленному в таблице.

2) Для своей программы придумайте минимум 5 тестов. 3) Протестируйте свою программу на этих тестах.

## Исходный код программы

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | **import** **re**  tests = ["artur-stepanov-03@mail.ru",  "sAtret@niuitmo.ru",  "timurzykov@gmail.com",  "dhfiheihgi@yandex.ru",  "oao@oaocom",  ".....@...",  "@.e"]  **for** testnumber **in** range(len(tests)):  str = tests[testnumber]  exemple = re.match( r'(.[a-zA-Z0-9\-\\_]\*)\@(.[a-zA-Z0-9\-\\_]\*)\.(.[a-zA-Z0-9\-\\_]\*)', str, re.M|re.I)  **if** exemple:  print(exemple.group(**2**),'.',exemple.group(**3**), sep="")  **else**:  print("Fail!") |

## Вывод программы

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | mail.ru  niuitmo.ru  gmail.com  yandex.ru  Fail!  Fail! |

# Вывод

Я понял, как использовать регулярные выражения и вспомнил синтаксис python.