

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Санкт-Петербургский национальный исследовательский университет ИТМО»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

ЛАБОРАТОРНАЯ РАБОТА №3

по дисциплине

‘ИНФОРМАТИКА’

‘Регулярные выражения’

Номер ИСУ 335989

Выполнил:

Студент группы Р3118

Павлов Александр

Сергеевич

Преподаватель:

Балакшин Павел

Валерьевич



УНИВЕРСИТЕТ ИТМО

Санкт-Петербург, 2021

Оглавление

Задания:.....	3
Коды:.....	4
Результаты работы программ:	11
Вывод:	13

Задания:

№1

- 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.
- 4) * нарисовав смайлик по вашему варианту при помощи средств языка программирования Python

Номер в ИСУ % 5	Глаза	Номер в ИСУ % 4	Нос	Номер в ИСУ % 7	Рот
0	:	0	-	0	(
1	;	1	<	1)
2	X	2	-{	2	o
3	8	3	<{	3	
4	=			4	\
				5	/
				6	P

№2

- 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. Пример тестов приведён в таблице.

1	<p>Довольно распространённая ошибка ошибка – это повтор слова. Вот в предыдущем предложении такая допущена. Необходимо исправить каждый такой повтор.</p> <p>Повтор это – слово, один или несколько пробельных символов, и снова то же слово.</p> <p>Пример:</p>	
	Ввод	Вывод
	Довольно распространённая ошибка ошибка – это лишний повтор повтор слова слова. Смешно, не не правда ли? Не нужно портить хор хоровод.	Довольно распространённая ошибка – это лишний повтор слова. Смешно, не правда ли? Не нужно портить хор хоровод.

№3

- 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.

1	<p>С помощью регулярного выражения найти в тексте слова, в которых встречается строго одна гласная буква (встречаться она может несколько раз). Пример таких слов: окно, трава, молоко, etc.</p> <p>После чего данные слова требуется отсортировать по увеличению длины слова.</p> <p>Пример:</p>	
	Ввод	Вывод
	Классное слово – обороноспособность, которое должно идти после слов: трава и молоко.	и слов идти слово трава должно молоко обороноспособность

Коды:

№1:

```
import re
```

```
import pygame
```

```
# =<| - смайлик
```

```
print('Номер ИСУ = 335989')
```

```

print('335989 % 5 =',335989%5,',','335989 % 4 =',335989%4,',','3354989 % 7
=','335989%7)

print('Мой смайлик - "=<|")

Test1 = '=<|=<|=<|=<|=<|=<|=<|=<|=<|=<|=<|=<|=<|=<|=<|=<|=<|=<|'
Test2 = '=<|=<|:-)=<X|=<|'
Test3 = 'X-{ O8<{P=<|=<'
Test4 = "
Test5 = '12341235123049878123047123074123087410237401923874-
12375r01347059124091237084517230846'

Mass = []

k = 1

Mass.append(Test1)
Mass.append(Test2)
Mass.append(Test3)
Mass.append(Test4)
Mass.append(Test5)

for i in Mass:

    result_regex = len(re.findall(r'=<\\|', i))

    result_python = i.count('=<|')

    counter = result_regex

    if result_regex == result_python:

        if counter % 100 >= 11 and counter % 100 < 20:

            print(f'Test{k} прошёл успешно, в тексте всего
{result_regex} смайликов "=<|".')

        elif counter % 10 == 1:

            print(f'Test{k} прошёл успешно, в тексте всего
{result_regex} смайлик "=<|".')

        elif counter % 10 == 2 or counter % 10 == 3 or counter % 10 == 4:

            print(f'Test{k} прошёл успешно, в тексте всего
{result_regex} смайлика "=<|".')

```

```

        else:
            print(f'Test{k} прошёл успешно, в тексте всего
{result_regex} смайликов "<|".')
        else:
            print(f'Test{k} не прошёл, автор сделал что-то не так, исправляй.')
        k += 1
print('Для рисовалки смайлика тыкните Enter')
_ = input()
s = 1
while s:
    pygame.init()
    keys = pygame.key.get_pressed()
    pygame.display.set_caption('Рисовалка смайлика')
    win = pygame.display.set_mode((700,500))
    pygame.draw.line(win,(255,255,255),(100,210),(250,210),7)
    pygame.draw.line(win,(255,255,255),(100,230),(250,230),7)
    pygame.draw.line(win,(255,255,255),(270,220),(420,150),7)
    pygame.draw.line(win,(255,255,255),(270,220),(420,290),7)
    pygame.draw.line(win,(255,255,255),(470,110),(470,330),7)
    for k in pygame.event.get():
        if k.type == pygame.QUIT:
            s = False
        if keys[pygame.K_ESCAPE]:
            s = False
    pygame.display.update()

```

№2:

```
import re
```

```
Test1 = 'Довольно распространённая ошибка ошибка – это лишний повтор  
повтор слова слова. Смешно, не не правда ли? Не нужно портить хор  
хоровод.'
```

```
Test2 = 'orange red blue green orange green blue'
```

```
Test3 = 'pirates ninjas cowboys ninjas pirates'
```

```
Test4 = 'привет привет как как дела дела. У меня нормально нормально, а у  
тебя?'
```

```
Test5 = '1234 1234 kak kak kak 1234 ghbdtn'
```

```
Massiv = []
```

```
Massiv.append(Test1)
```

```
Massiv.append(Test2)
```

```
Massiv.append(Test3)
```

```
Massiv.append(Test4)
```

```
Massiv.append(Test5)
```

```
k = 1
```

```
for p in Massiv:
```

```
    if len(p) != 0:
```

```
        p_regex = p
```

```
        p_python = p
```

```
        result = re.findall(r'\b(\w+)\s\1\b',p_regex)
```

```
        for i in result:
```

```
            p_regex = re.sub(i,"",p_regex,1)
```

```
            p_regex = re.sub(r'\s{2}',' ',p_regex)
```

```
            p_regex = re.sub(r'^\s',"",p_regex)
```

```
        massiv = p_python.split(' ')
```

```

flag = 0
for i in range(len(massiv)-1):
    if massiv[i][-1] in '?!,-' and len(massiv[i]) >= 2:
        temp = massiv[i][:-1]
        flag = 1
    if massiv[i] == massiv[i+1]:
        massiv.pop(i)
        massiv.append(' ')
    if flag == 1 and massiv[i-1] == temp:
        massiv.pop(i-1)
        massiv.append(' ')
        flag = 0

a = ""
for i in massiv:
    if i != ' ':
        a += i + ' '

if (a == p_regex + ' '):
    print(f'Test{k} прошёл успешно, автор молодец.')
    print(a)
else:
    print(f'Test{k} прошёл некруто, автор переделывай.')
    print(a)
    print(p_regex)

k += 1

else:
    print('Ты меня не обманешь!(строки нет)')
    k += 1

```


№3:

```
import re
```

```
Test1 ='Классное слово – Обороноспособность, которое должно идти после  
слов: трава и молоко.'
```

```
Test2 = 'На самом деле я абсолютно не знаю. Что здесь писать?'
```

```
Test3 = "
```

```
Test4 = '123 123 123'
```

```
Test5 = 'Я.Абсолютно.Не.Имею.фантазии.дЛя.таках.дЕл.'
```

```
Massiv = []
```

```
Massiv.append(Test1)
```

```
Massiv.append(Test2)
```

```
Massiv.append(Test3)
```

```
Massiv.append(Test4)
```

```
Massiv.append(Test5)
```

```
count = 1
```

```
for chto in Massiv:
```

```
    test = "
```

```
    for i in chto:
```

```
        if ord(i) >= 1040 and ord(i) <= 1071:
```

```
            i = chr(ord(i) + 32)
```

```
        test += i
```

```
result = re.findall(r'\b(\w*[аиеёоуыэюя]\w*)\b',test)
```

```
result_regex = []
```

```
for i in result:
```

```
    p = re.search(r'a|и|е|ё|о|у|ы|э|ю|я',i).group(0)
```

```
    p = re.sub(p,"",i)
```

```

p = re.findall(r'\b(\w*[аиеёоуыэюя]\w*)',p)
if len(p) == 0:
    result_regex.append(i)
counter = 1
for k in range(len(result_regex)):
    for j in range(k,(len(result_regex))):
        if len(result_regex[j]) < len(result_regex[k]):

result_regex[j],result_regex[k]=result_regex[k],result_regex[j]
#проверка обычными средствами путхона
massiv_obichn = []
result_python = []
a = "
for i in test:
    for j in i:
        if ord(j) >= 1072 and ord(j) <= 1103:
            a += j
        else:
            if a != "":
                if (a.count('a') + a.count('и') + a.count('е') +
a.count('ё') + a.count('о') + a.count('у') + a.count('ы') + a.count('э') + a.count('ю') +
a.count('я')) >= 1:

                                massiv_obichn.append(a)

            a = "
for i in range(len(massiv_obichn)):
    for j in massiv_obichn[i]:
        temp = massiv_obichn[i]
        if (j == 'a' or j == 'и' or j == 'е' or j == 'ё' or j == 'о' or j == 'у' or
j == 'ы' or j == 'э' or j == 'ю' or j == 'я'):

```

```

        temp = temp.replace(j, "")

        if (temp.count('a') + temp.count('и') + temp.count('е') +
temp.count('ё') + temp.count('о') + temp.count('у') + temp.count('ы') +
temp.count('э') + temp.count('ю') + temp.count('я')) == 0:

            result_python.append(massiv_obichn[i])

            break

for k in range(len(result_python)):
    for j in range(k,(len(result_python))):
        if len(result_python[j]) < len(result_python[k]):

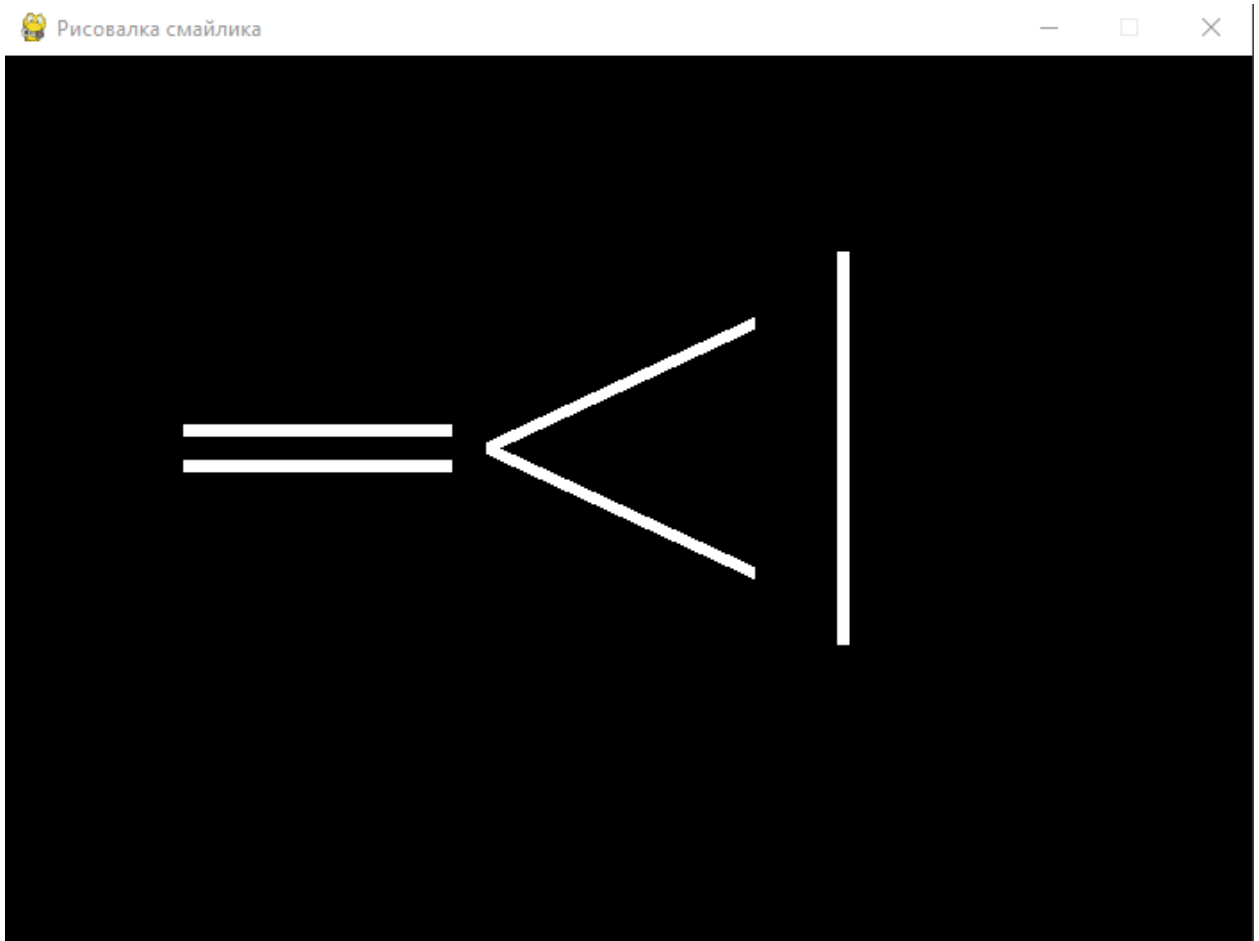
result_python[j],result_python[k]=result_python[k],result_python[j]
if result_python == result_regex and len(result_python) != 0:
    print(f'Test{count} прошёл успешно, автор молодец!')
    print('Ответ:',result_python)
elif result_regex == result_python and len(result_python) == 0:
    print(f'Test{count} прошёл успешно, ответ - пустая строка.')
else:
    print(f'Test{count} лоханулся, переделывать пора.')
    print('Обычный ответ:', result_python)
    print('Ответ через регулярки:',result_regex)

count += 1

```

Результаты работы программ:
№1

```
Номер ИСУ = 335989
335989 % 5 = 4 ; 335989 % 4 = 1 ; 3354989 % 7 = 3
Мой смайлик - "=<|"
Test1 прошёл успешно, в тексте всего 21 смайлик "=<|".
Test2 прошёл успешно, в тексте всего 3 смайлика "=<|".
Test3 прошёл успешно, в тексте всего 1 смайлик "=<|".
Test4 прошёл успешно, в тексте всего 0 смайликов "=<|".
Test5 прошёл успешно, в тексте всего 0 смайликов "=<|".
Для рисовалки смайлика тыкните Enter
```



№2:

Test1 прошёл успешно, автор молодец.

Довольно распространённая ошибка – это лишний повтор слова. Смешно, не правда ли? Не нужно портить хор хоровод.

Test2 прошёл успешно, автор молодец.

orange red blue green orange green blue

Test3 прошёл успешно, автор молодец.

pirates ninjas cowboys ninjas pirates

Test4 прошёл успешно, автор молодец.

привет как дела. У меня нормально, а у тебя?

Test5 прошёл успешно, автор молодец.

1234 kak kak 1234 ghbdtn

№3:

Test1 прошёл успешно, автор молодец!

Ответ: ['и', 'слов', 'идти', 'трав', 'слово', 'должно', 'молоко', 'обороноспособность']

Test2 прошёл успешно, автор молодец!

Ответ: ['я', 'на', 'не', 'что', 'деле', 'здесь']

Test3 прошёл успешно, ответ - пустая строка.

Test4 прошёл успешно, ответ - пустая строка.

Test5 прошёл успешно, автор молодец!

Ответ: ['я', 'не', 'для', 'дел', 'таках']

Вывод:

В ходе выполнения данной лабораторной работы я научился работать с регулярными выражениями и применять их на практике.