Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики

Факультет программной инженерии и компьютерной техники

**Лабораторная работа #4**

**"Регулярные выражения"**

Вариант 8

Выполнил:

Студент 1-го курса, группы P3114

Ермаков Т. С., 311733

Преподаватели:

Балакшин Павел Валерьевич

Ильина Аглая Геннадьевна

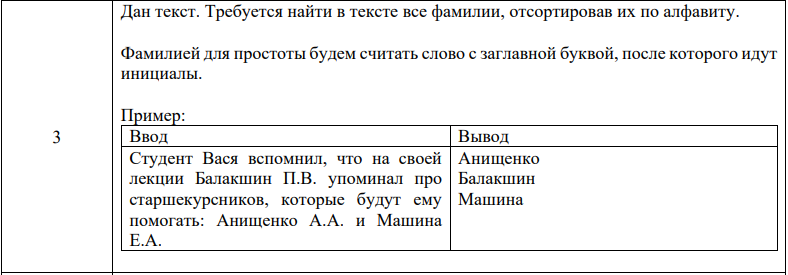
Санкт-Петербург, 2020

**Порядок** **выполнения** **работы**

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.



**Выполнение работы**

**script1.0.py**

import re  
pattern = re.compile('[А**-**Я][а**-**я]\* [А**-**Я].[А**-**Я].')  
text = ["Студент Вася вспомнил, что на своей лекции Балакшин П.В. упоминал про старшекурсников, которые будут ему помогать: Анищенко А.А. и Машина Е.А.",  
 "Морозов Н.С. не знает, что на лекции сказал Балакшин П.В., так как он уснул на паре из-за того, что вчера много занимался и поздно лёг спать",  
 "Сегодня на лабораторной по информатике (преподаватель Ильина А.С.) защитились все студенты, в том числе и Ермаков Т.С.",  
 "Группа P3114: Авдеев С.С., Бейгер Г.Ю., Возмилов З.Е., Гадеев Р.Р., Голованова Д.В., ...",  
 "Путин В.В. - молодец))"]  
for s in text:  
 matches = pattern.findall(s)

matches.sort()  
 for i in matches:  
 print(i[0:-5])  
 print()

**Вывод:**

Балакшин

Анищенко

Машина

Морозов

Балакшин

Ильина

Ермаков

Авдеев

Бейгер

Возмилов

Гадеев

Голованова

Путин

**Доп. задание №1**

8 % 4 = 0

Вывести все предложения из Гамлета, в которых ровно 6 слов, одно из которых двусложное. Придумать способ, как проверить корректность работы регулярного выражения и проверить.

**script1.1.py**

import re  
v = 'aeiouyAEIOUY'  
w = '^\s.!?'  
with open('Hamlet.txt', 'r') as f:  
 text = f.read()  
for match in re.finditer(rf'[A**-**Z]([{w}]+(?!\n\s\S)\s+){{5}}[{w}]+[.!?]', text):  
 s = match.group()  
 if re.search(rf'[\s.!?][{w}{v}]\*([{v}][{w}{v}]\*){{2}}[\s.!?]', s):  
 print(' '.join(s.split()))

**Вывод:**

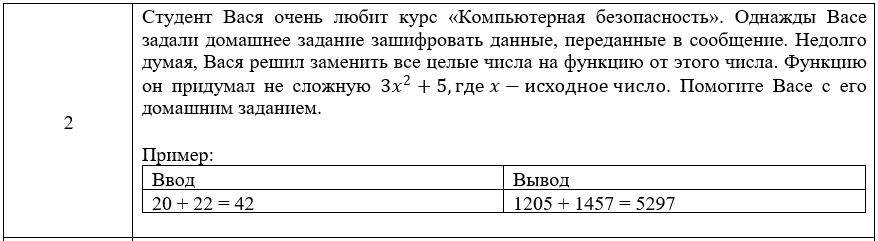
В файле result.txt

**Проверка:**

Заметим, что предложение из 6 слов где-то в среднем 35 символов (мин. длина 22, макс. длина 55). Составим более простое и понятное регулярное выражение, которое проверяет предложения на эту длину и найдём их (получится неточно, но зато мы поймем, правильно всё получилось или нет).

**Доп. задание №2**

**8 % 6 = 2**

****

**script1.2.py**

import re  
s = input()  
pattern = re.compile('\d+[.,]{0}')  
original = pattern.findall(s)  
for i in original:  
 n = int(i)  
 n = 3 \* n \* n + 5  
 s = s.replace(i, str(n))  
print(s)

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| 20 + 22 = 42 | 1205 + 1457 = 5297 |
| 15.5 + 5.5 = 21 | 15.5 + 5.5 = 1328 |
| 15,5 + 5,5 = 21 | 15,5 + 5,5 = 1328 |
| привет 123 hello | привет 45392 hello |
| 100 баллов | 30005 баллов |

**Вывод**

В результате выполнения лабораторной работы я научился составлять регулярные выражения, применять их на практике в программах на Python. Благодаря им можно находить в больших текстах нужные данные и сокращать код, занимающийся его парсингом, однако недостатком является их запутанный вид.

**Ресурсы**

<https://regex101.com/>

<https://vk.com/doc-31201840_572834504>

<https://vk.com/video-31201840_456239046>

<https://habr.com/ru/post/349860/>

<https://tproger.ru/translations/regular-expression-python/>

<http://website-lab.ru/article/regexp/shpargalka_po_regulyarnyim_vyirajeniyam/>