A close up of a logo

Description automatically generated

Факультет ПИиКТ

Лабораторная работа №3 по информатике

Регулярные выражения

Вариант 291011

**Выполнил:**

Давааням Баясгалан

группа P3111

**Преподаватель:**

Малышева Татьяна Алексеевна

г. Санкт-Петербург

2021 год

Оглавление

[Задание на 60 баллов (Смайлики) 3](#_Toc86086913)

[Код программы на Python 3](#_Toc86086914)

[Результат программы 4](#_Toc86086915)

[Дополнительное задание №1 (+18 баллов) 5](#_Toc86086916)

[Код программы на Python 5](#_Toc86086917)

[Результат программы 6](#_Toc86086918)

[Дополнительное задание №2 (+22 баллов) 7](#_Toc86086919)

[Код программы на Python 7](#_Toc86086920)

[Результат программы 8](#_Toc86086921)

[Вывод 9](#_Toc86086922)

[Список литературы 9](#_Toc86086923)

# Задание на 60 баллов (Смайлики)

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].

|  |  |  |
| --- | --- | --- |
| [глаза] | [нос] | [рот] |
| 1 | 3 | 0 |
| ; | <{ | ( |

## Код программы на Python

import re

def solve(x):

    return len(re.findall(r";<{\(",x)) # найти все смайлики по структуре,

добавить в лист и возвращать длину листа.

address\_input = "D:\ITMO\\1st Semester\Computer Science (Basics)\lab3\lab3\_1\input.txt"

address\_output = 'D:\ITMO\\1st Semester\Computer Science (Basics)\lab3\lab3\_1\output.txt'

input\_file = open(address\_input,'r',encoding='utf-8')

output\_file = open(address\_output,'w',encoding='utf-8')

x = input\_file.read().split('\n')

for i in range(len(x)):

    output\_file.write(str(solve(x[i]))+'\n')

input\_file.close()

output\_file.close()

## Результат программы

|  |  |
| --- | --- |
| Ввод | Вывод |
| ;<{(;<{(;<{(;<{(;<{(;<{(;<{(;<{(;<{(;<{(;<{(;<{( | 12 |
| ;<{(;<{(;<{(;<{(;<{(;<{(;<{(;<{(;<{(;<{(;<{(;<{(;<{(;<{(;<{( | 15 |
| ;<{{(;:<{(;<{(;<{(;<{a(;<{(;<{(;<{(;<{(;<{(;<{(;<{a(;<{(;<{(;<{( | 11 |
| ;<{(;<<{(;<{(;<{{(;<{( | 3 |
| В ИМТО учатся;<{(;<{{(; более 10000 <{(;<{(;:<{( студентов. | 2 |
| ITMO ;<University ;<{( is a state-;<{(supported university ;<{(in Saint<{( Petersburg;<{( | 4 |
| ITMO Univer;<{( sity has an ;<{( acceptance ;{(rate of 15%, enrollment - 12500;<{( | 3 |

# Дополнительное задание №1 (+18 баллов)

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. см на *Рисунок 1*

Table

Description automatically generated

Рисунок 1

## Код программы на Python

import re

def solve(x):

    string = ''

    a = re.split(r'\W+',x) #разделить по любой не-буква, не-цифра и не подчёркивание и добавить в лист, (+) значит может более синоним

    for i in range(len(a)):

        if(i<len(a)-1): #если это не последный элемент листа

            match = re.search(r'\w\*[УуЕеЫыАаОоЭэЯяИиЮю]{2}\w\*',a[i])#проверить это слово,в которых две гласные стоят подряд

            if(match and len(re.findall(r'[йЦцКкНнГгШшЩщЗзХхФфВвПпРрЛлДдЖжБбТтМмСсЧч]',a[i+1]))<4): #проверить после этого слово идет ли слово, в котором не больше 3 согласных

                string += a[i] + ' '

    return string

address\_input = "D:\ITMO\\1st Semester\Computer Science (Basics)\lab3\lab3\_2\input.txt"

address\_output = 'D:\ITMO\\1st Semester\Computer Science (Basics)\lab3\lab3\_2\output.txt'

input\_file = open(address\_input,'r',encoding='utf-8')

output\_file = open(address\_output,'w',encoding='utf-8')

x = input\_file.read().split('\n')

for i in range(len(x)):

    output\_file.write(str(solve(x[i]))+'\n')

input\_file.close()

output\_file.close()

## Результат программы

|  |  |
| --- | --- |
| Ввод | Вывод |
| Кривошеее существо гуляет по парку | гуляет |
| Для своей программы придумайте минимум пять тестов |  |
| Сжатие газа вызывает нагревание газа | Сжатие нагревание |
| В процессе горения могут образовываться газы | горения |
| Испарение играет важную роль в паровых машинах, в двигателях внутреннего сгорания | Испарение играет важную |
| При падении тел с небольшой высоты движение происходит по вертикальной прямо | падении происходит |

# Дополнительное задание №2 (+22 баллов)

1. Реализуйте программный продукт на языке Python, используя регулярные выражения по варианту, представленному в таблице.
2. Реализуйте программный продукт на языке Python, используя регулярные выражения по варианту, представленному в таблице.
3. Протестируйте свою программу на этих тестах.

Пример тестов приведён в таблице. см на *Рисунок 2*

Table

Description automatically generated

Рисунок 2

## Код программы на Python

import re

def solve(x):

    string = ''

    a = re.split(r'\W+',x) #разделить по любой не-буква, не-цифра и не подчёркивание и добавить в лист, (+) значит может более синоним

    if(a[-1] == 'P3111'): #проверить номер группы

        b = re.findall(r'[А-ЯЁ]',x) #найти всю большаую букву и добавить в лист

        if(b[0] != b[1] or b[1] != b[2] or b[0] != b[2]): #если они не равны

друг с другом, то записать в файл

            string += x

            output\_file.write(string+'\n')

    else: #если номер группы отличается от "P3111", то записать в файл

        string += x

        output\_file.write(string+'\n')

address\_input = "D:\ITMO\\1st Semester\Computer Science (Basics)\lab3\lab3\_3\input.txt"

address\_output = 'D:\ITMO\\1st Semester\Computer Science (Basics)\lab3\lab3\_3\output.txt'

input\_file = open(address\_input,'r',encoding='utf-8')

output\_file = open(address\_output,'w',encoding='utf-8')

x = input\_file.read().split('\n')

for i in range(len(x)):

    solve(x[i])

input\_file.close()

output\_file.close()

## Результат программы

|  |  |
| --- | --- |
| Ввод | Вывод |
| Горбунов В.В P3111  Савин С.С. P3111  Журавлёв Р.Г. P3111  Селезнёв С.С P3211 | Горбунов В.В P3111  Журавлёв Р.Г. P3111  Селезнёв С.С P3211 |
| Лаврентьев Л.Л P31113  Поляков Л.А. P3212  Баранов Б.Б P3111  Лапин О.Я. P3111 | Лаврентьев Л.Л P31113  Поляков Л.А. P3212  Лапин О.Я. P3111 |
| Уваров С.Р. P3211  Щербаков М.Щ P3111  Баранов Б.Б P3111  Селезнёв С.С P3211 | Уваров С.Р. P3211  Щербаков М.Щ P3111  Селезнёв С.С P3211 |
| Лапин О.Я. P3111  Лаврентьев Л.Л P31113  Баранов Б.Б P3113  Савин С.С. P3111 | Лапин О.Я. P3111  Лаврентьев Л.Л P31113  Баранов Б.Б P3113 |
| Селезнёв С.С P33111  Щербаков М.Щ P3111  Уваров С.Р. P3111  Баранов Б.Б P3111 | Селезнёв С.С P33111  Щербаков М.Щ P3111  Уваров С.Р. P3111 |

# Вывод

При выполнении лабораторной работы я ознакомился с синтакмом Python. Так же я научил как использовать регулярное выражение. Еще научил работать с файлами.

# Список литературы

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

<https://en.wikipedia.org/wiki/Regular_expression>