

Федеральное государственное автономное образовательное

учреждение высшего образования

«Национальный исследовательский университет ИТМО»

**Отчёт по лабораторной работе 3**

Тема: Регулярные выражения

Выполнил**:** студент группы Р3115 Храбров Артём Алексеевич

Проверил**:** Белокон Юлия Алексеевна

Дата сдачи: 21.10.2024

2024

**Содержание**

1. Задание 1
2. Задание 2
3. Задание 3
4. Заключение

**Задание 1**

***Задание:***

Написать программу на Python с использованием регулярных выражений, которая будет считать количество смайликов =-( в тексте.

***Решение:***

Для теста программы используется файл tests/test\_1.txt. Для получения данных из него используется функция load\_tests из файла loader.py. В неё передаётся ссылка на файл. В файле test\_1.txt тесты отелены друг от друга пустой строкой. При необходимости можно добавить в файл ещё тестов.

***Код программы:***

**import** re

**from** loader **import** load\_tests

# Загрузка тестов

tests **=** load\_tests**(**"tests/test\_1.txt"**)**

# Подсчёт количества смайликов

**for** i **in** **range(len(**tests**)):**

rez **=** re**.**findall**(**r"=-\("**,** tests**[**i**])**

**print(**f"Тест {i**+**1}: {**len(**rez**)**} смайликов '=-('"**)**

***Содежимое файла test\_1:***

test1=-2==-313---=-=-)

ret;', ;:)(@@ "":=-(

=-()-=()12ds++-~@fd=-(=-

test2-,.=-=-=(= -()))$231=-)=-(

=-(====- (=--,.(=-))({}

test3WWWPX.com<><>/sfdjkDSDDS

PETRU\_\_\_+-das=-d=as-mksf=-)

=-(test4=-)jj =- (=- ( -=-(

test5,,\*+-\*-/@@#Y$@(&(#)!@\_\_:SDDQI=-(

***Правильный ответ и вывод программы:***

1. 3
2. 2
3. 0
4. 2
5. 1

Тест 1: 3 смайликов '=-('

Тест 2: 2 смайликов '=-('

Тест 3: 0 смайликов '=-('

Тест 4: 2 смайликов '=-('

Тест 5: 1 смайликов '=-('

**Задание 2**

***Задание:***

Написать программу на Python с использованием регулярных выражений, где заменить в тексте все вхождения времени в формате HH:MM:SS или HH:MM на строку (TBD). HH от 00 до 23; MM от 00 до 59; SS от 00 до 59

***Решение:***

Для теста программы используется файл tests/test\_2.txt. Для получения данных из него используется функция load\_tests из файла loader.py. В неё передаётся ссылка на файл. В файле test\_2.txt тесты отелены друг от друга пустой строкой. При необходимости можно добавить в файл ещё тестов.

***Код программы:***

**import** re

**from** loader **import** load\_tests

# Загрузка тестов

tests **=** load\_tests**(**"tests/test\_2.txt"**)**

# Замена всех вхождений времени на строку (TBD)

**for** i **in** **range(len(**tests**)):**

rez **=** re**.**sub**(**r"(([01][0-9])|(2[0-3])):[0-5][0-9](:[0-5][0-9])?"**,** "(TBD)"**,** tests**[**i**])**

**print(**f"ТЕСТ {i**+**1}:\n{rez}"**)**

***Содежимое файла test\_2:***

Заявления на отчисление ПСЖ принимаются с 09:00

PS: с 16:20:01

С 03.03.2025 лекции по предмету "История русской культуры"

будет вести Махаммад Бертнджи Бекузбекович.

Вопросы о переводе не рассматриваются в аудитории 666 с 7:00 до 09:00:00

Просто время, мне лень придумывать. 12:00:60 23:33

52:98:13 DDCX03:40:33BB 0320: 00

12:09:99 12:21 12: 12 9:33

33:33:33 15:34:00 00:00:00 00:00 10:01 :01:0

***Вывод программы:***

ТЕСТ 1:

Заявления на отчисление ПСЖ принимаются с (TBD)

PS: с (TBD)

ТЕСТ 2:

С 03.03.2025 лекции по предмету "История русской культуры"

будет вести Махаммад Бертнджи Бекузбекович.

Вопросы о переводе не рассматриваются в аудитории 666 с 7:00 до (TBD)

ТЕСТ 3:

Просто время, мне лень придумывать. (TBD):60 (TBD)

52:98:13 DDCX(TBD)BB 0320: 00

ТЕСТ 4:

(TBD):99 (TBD) 12: 12 9:33

ТЕСТ 5:

33:33:33 (TBD) (TBD) (TBD) (TBD) :01:0

**Задание 3**

***Задание:***

Написать программу на Python с использованием регулярных выражений, которая проверит правильность введённого email адреса. В случае верности адреса она должна выдать почтовый сервер адреса, иначе «Fail!». Почтовый адрес содержит только буквы, цифры, «\_», «.». Почтовый сервер только буквы и «.». Причём почтовый сервер обязательно должен содержать домен верхнего уровня (.com .ru и тд.).

***Решение:***

Для теста программы используется файл tests/test\_3.txt. Для получения данных из него используется функция load\_tests из файла loader.py. В неё передаётся ссылка на файл. В файле test\_3.txt тесты отелены друг от друга пустой строкой. При необходимости можно добавить в файл ещё тестов.

***Код программы:***

**import** re

**from** loader **import** load\_tests

# Загрузка тестов

tests **=** load\_tests**(**"tests/test\_3.txt"**)**

# Проверка правильности адреса и вывод результата

**for** i **in** **range(len(**tests**)):**

**if** re**.**fullmatch**(**r"^[\w.]+@[a-zA-Z.]+(\.[a-zA-Z]+)$"**,** tests**[**i**].**strip**()):**

**print(**f"Тест {i **+** 1}: {re**.**search**(**r"@[\w.]+"**,** tests**[**i**])[**0**][**1**:]**}"**)**

**else:**

**print(**f"Тест {i**+**1}: Fail!"**)**

***Содежимое файла test\_3:***

prosto\_pochta@gmail.com

ochen\_dlinnaya\_pohta.bigfish.azino777@razvodloxov.com

898-tuper@gmail.com

Valentin\_Gennadievich\_1952@odnoklassniki.ru

-Petr228@mail.ru

***Вывод программы:***

Тест 1: gmail.com

Тест 2: razvodloxov.com

Тест 3: Fail!

Тест 4: odnoklassniki.ru

Тест 5: Fail!

**Заключение**

В ходе лабораторной работы я познакомился с регулярными выражениями, научился использовать регулярные выражения в Python. Применил их для решения нескольких простых задач, самостоятельно протестировал правильность работы программы. Написал вывод.