НИУ ИТМО

Факультет программной инженерии и компьютерной техники

Отчет по Лабораторной работе №3    
по курсу “Информатика”

"Регулярные выражения"    
Вариант №335038

Выполнила:

Студентка группы P3110

Бегинина Анастасия Алексеевна   
Преподаватели:

Балакшин Павел Валерьевич  
Рудникова Тамара Владимировна

Санкт-Петербург

2021

Оглавление

[Задания 3](#_Toc84983596)

[1. Задание на 60 баллов (Смайлики) 3](#_Toc84983597)

[2. Доп. задание No1 (+18 баллов) 3](#_Toc84983598)

[3. Доп. задание No2 (+22 баллов) 3](#_Toc84983599)

[Основные этапы вычисления 4](#_Toc84983600)

[Вывод 5](#_Toc84983601)

[Список литературы 6](#_Toc84983602)

[Приложения 7](#_Toc84983603)

# Задания

## 1. 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*].  
Вариантом является различные наборы глаз, носов и ртов. (см. Приложение 1)

*Пример смайлика:* 8<{P  
4) **\*** нарисовав смайлик по вашему варианту при помощи средств языка программирования Python, можно заработать дополнительные баллы.

## 2. Доп. задание No1 (+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. Приложение 2)

## 3. Доп. задание No2 (+22 баллов)

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.

(см. Приложение 3)

# Основные этапы вычисления

1.

def smile(message):

mySmile = '8-{\\'

return (message.count(mySmile))

print(smile('message 8-{\\')) #answer 1

print(smile('message 8-{<\ ^&))) 8&\*&(&(\*&^(\* ')) #answer 0

print(smile('dkfmdlkmfdlkmfldk 8-{\\kjijsdiojads lkajdlkaj 8-{\\ iakijsdkjnaskjnxaskjdh 8-{\\')) #answer 3

print(smile('8-{\\8-{\\8-{\\8-{\\8-{\\ 8-{\\')) #answer 6

(см. Приложение 4)

2.

import re

def time\_pattern(message): #HH:MM or HH:MM:SS

pattern1 = r"(([0-1]{1}[0-9]{1})|([2]{1}[0-3]{1})):[0-5][0-9]:[0-5][0-9]"

pattern2 = r"(([0-1]{1}[0-9]{1})|([2]{1}[0-3]{1})):[0-5][0-9]"

res = (re.sub(pattern1, "(TBD)", message, count=0))

res = (re.sub(pattern2, "(TBD)", res, count=0))

return res

print(time\_pattern("12:20:20")) #time is correct. (TBD)

print(time\_pattern("35:20:20")) #time is incorrect. hours should be less than 24. 35:20:20

print(time\_pattern("23:20:20")) #time is correct. (TBD)

print(time\_pattern("23:20 13:10 16:00:00 35:00 12:78:00 13:79")) #(TBD) (TBD) (TBD) 35:00 12:78:00 13:79

print(time\_pattern("Hello it's 12:20 o'clock. Specifically 12:20:09")) #Hello it's (TBD) o'clock. Specifically (TBD)

(см. Приложение 5)

3.

import re

def find\_numbers(message):

pattern = r'\d+'

answer = message[::]

while (re.search(pattern, message) != None):

first = int(re.search(pattern, message).group(0))

second = 3 \* (first \*\* 2) + 5

message = message.replace(str(first), '')

answer = answer.replace(str(first), str(second))

return answer

print(find\_numbers("20 + 22 = 42")) #1205 + 1457 = 5297

print(find\_numbers("shjdckljsadnclksdmc")) #shjdckljsadnclksdmc

print(find\_numbers("1234567890")) #4572473625057156305

print(find\_numbers("1 + 2 + 3 = 6")) #8 + 17 + 32 = 113

print(find\_numbers("shjdckljsadnclksdmc 0 0 0 0 1525 ")) #shjdckljsadnclksdmc 5 5 5 5 6976880

(см. Приложение 6)

# Вывод

В ходе выполнения лабораторной работы я научилась работать с регулярными выражениями в Python. Узнала об основах синтаксиса, тонкостях экранирования и перечислениях.

# Список литературы

1. Регулярные выражения в Python от простого к сложному. Подробности, примеры, картинки, упражнения // habr.com: блог об IT. 2007 URL: https://habr.com/ru/post/349860/ (дата обращения: 13.10.2021).
2. Регулярные выражения (regex) - основы. // habr.com: блог об IT. 2007 URL: https://habr.com/ru/post/545150/ (дата обращения: 13.10.2021).

# Приложения

Приложение 1.

Table

Description automatically generated

Приложение 2.

Table

Description automatically generated

Приложение 3.

Table

Description automatically generated

Приложение 4.

A picture containing text

Description automatically generated

Приложение 5.Text

Description automatically generated

Приложение 6.

Graphical user interface, text, application

Description automatically generated