Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет Программной инженерии и компьютерной техники

Информатика

Отчёт по лабораторной работе №3

Регулярные выражения

Вариант № 2

Выполнил: студент группы P3214

Серов А.А.

Преподаватель: Балакшин П.В.

Санкт-Петербург

2025

**Задание:**

**Дополнительное задание 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. Все 5 тестов необходимо показать при защите. Пример тестов приведён в таблице.

3) Можно использовать циклы и условия, но основной частью решения должны быть регулярные выражения.

**Дополнительное задание 2:**

1) Реализуйте программный продукт на языке Python, используя регулярные выражения по варианту, представленному в таблице.

2) Для своей программы придумайте минимум 5 тестов. Все 5 тестов необходимо показать при защите.

3) Протестируйте свою программу на этих тестах.

4) Можно использовать циклы и условия, но основной частью решения должны быть регулярные выражения.

*Вариант: 470162 % 6 =* ***2***

**Задание 1:**

**Программа:**

import re

def extract\_surnames(text):

pattern = r'\b([A-ZА-ЯЁ][a-zа-яё-]+(?:-[A-ZА-ЯЁ][a-zа-яё-]+)?)(?:\s[A-ZА-ЯЁ]\.){1,2}\b'

matches = re.findall(pattern, text)

surnames = set(matches)

sorted\_surnames = sorted(surnames)

return sorted\_surnames

# Тесты

tests = [

"Студент Вася вспомнил, что на своей лекции Балакшин П.В. упоминал про старшекурсников, которые будут ему помогать: Анищенко А.А., Машина Е.А. и Голованова-Иванова Д.В.",

"На встрече присутствовали Иванов И.И., Петров П.П. и Сидоров С.С.",

"В этом году на олимпиаде победили Смирнов А.А. и Кузнецов Б.Б.",

"В классе учатся: Ковалев А.А., Сидорова М.М., и Петрова Н.Н.",

"На конференции выступали: Лебедев А.А., Лебедев-Лебедева В.В. и Лебедева Б.Б"

]

# Ожидаемые результаты

expected\_results = [

['Анищенко', 'Балакшин', 'Голованова-Иванова', 'Машина'],

['Иванов', 'Петров', 'Сидоров'],

['Кузнецов', 'Смирнов'],

['Ковалев', 'Петрова', 'Сидорова'],

['Лебедев', 'Лебедев-Лебедева', 'Лебедева']

]

# Проверка тестов

for i, test in enumerate(tests):

result = extract\_surnames(test)

print(f"Тест {i + 1}:")

print(f"Входные данные: {test}")

print(f"Результат: {result}")

print(f"Ожидаемый результат: {expected\_results[i]}")

print(f"Тест {'пройден' if result == expected\_results[i] else 'не пройден'}\n")

**Тесты и их результаты:**

Тест 1:

Входные данные: Студент Вася вспомнил, что на своей лекции Балакшин П.В. упоминал про старшекурсников, которые будут ему помогать: Анищенко А.А., Машина Е.А. и Голованова-Иванова Д.В.

Результат: ['Анищенко', 'Балакшин', 'Голованова-Иванова', 'Машина']

Ожидаемый результат: ['Анищенко', 'Балакшин', 'Голованова-Иванова', 'Машина']

Тест пройден

Тест 2:

Входные данные: На встрече присутствовали Иванов И.И., Петров П.П. и Сидоров С.С.

Результат: ['Иванов', 'Петров', 'Сидоров']

Ожидаемый результат: ['Иванов', 'Петров', 'Сидоров']

Тест пройден

Тест 3:

Входные данные: В этом году на олимпиаде победили Смирнов А.А. и Кузнецов Б.Б.

Результат: ['Кузнецов', 'Смирнов']

Ожидаемый результат: ['Кузнецов', 'Смирнов']

Тест пройден

Тест 4:

Входные данные: В классе учатся: Ковалев А.А., Сидорова М.М., и Петрова Н.Н.

Результат: ['Ковалев', 'Петрова', 'Сидорова']

Ожидаемый результат: ['Ковалев', 'Петрова', 'Сидорова']

Тест пройден

Тест 5:

Входные данные: На конференции выступали: Лебедев А.А., Лебедев-Лебедева В.В. и Лебедева Б.Б

Результат: ['Лебедев', 'Лебедев-Лебедева', 'Лебедева']

Ожидаемый результат: ['Лебедев', 'Лебедев-Лебедева', 'Лебедева']

Тест пройден

*Вариант: 470162 % 8 =* ***2***

**Задание 2:**

**Программа:**

import re

def extract\_bitcoin\_price(html\_content):

    pattern = r'Цена Bitcoin в реальном времени сегодня составляет ₽([\d,]+\.\d{2}) RUB'

    match = re.search(pattern, html\_content)

    if match:

        return match.group(1)

    else:

        return None

def run\_tests():

    test\_cases = [

        {

            "ввод": '''<meta name="daily\_volume" content="В суточным объемом торгов ₽2,835,029,974,960.63 RUB."/>

                        <meta name="daily\_price" content="Мы обновляем нашу цену BTC к RUB в режиме реального времени."/>

                        <meta name="daily\_price" content=" Цена Bitcoin в реальном времени сегодня составляет ₽5,797,806.88 RUB."/>

                        <meta name="daily\_price" content="Ethereum стоит на данный момент ₽229,590,78 RUB."/>''',

            "вывод": "5,797,806.88"

        },

        {

            "ввод": '''<meta name="daily\_price" content="Цена Bitcoin в реальном времени сегодня составляет ₽1,234,567.89 RUB."/>''',

            "вывод": "1,234,567.89"

        },

        {

            "ввод": '''<meta name="daily\_price" content="Цена Bitcoin в реальном времени сегодня составляет ₽0.01 RUB."/>''',

            "вывод": "0.01"

        },

        {

            "ввод": '''<meta name="daily\_price" content="Цена Bitcoin в реальном времени сегодня составляет ₽10,000,000.00 RUB."/>''',

            "вывод": "10,000,000.00"

        },

        {

            "ввод": '''<meta name="daily\_price" content="Ethereum стоит на данный момент ₽229,590,78 RUB."/>''',

            "вывод": None  # Нет информации о Bitcoin

        }

    ]

    for i, test\_case in enumerate(test\_cases):

        result = extract\_bitcoin\_price(test\_case["ввод"])

        assert result == test\_case["вывод"], f"Тест {i+1} не пройден: вывод {test\_case['вывод']}, ответ {result}"

        print(f"Тест {i + 1} пройден.")

run\_tests()

**Тесты и их результаты:**

**Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание**

Тест 1 пройден.

Тест 2 пройден.

Тест 3 пройден.

Тест 4 пройден.

Тест 5 пройден.

**Вывод:**

В ходе проделанной лабораторной работы мы на примере интересных задач научились применять регулярные выражения в python при помощи соответствующей библиотеки (модуля) re, а также убедились, что программа работает исправно посредством проведения её тестов.