Федеральное государственное автономное образовательное учреждение

высшего образования «Национальный исследовательский университет

ИТМО»

*Факультет программной инженерии и компьютерной техники*

*Направление подготовки: 09.03.04 - Нейротехнологии и программирование*

*Дисциплина «Информатика»*

**Отчет**

**По лабораторной работе №3. «Регулярные выражения и**

**языки разметки документов. Python»**

Выполнил:

Мухин Никита Денисович

Группа: Р3123

Преподаватель:

Болдырева Елена Александровна

Г. Санкт-Петербург, 2023 г

**Оглавление**

[Задание 2](#_Toc114643031)

[Выполнение работы](#_Toc114643032) 4

[Заключение 10](#_Toc114643035)

[Список литературы.](#_Toc114643036) 11

# **Задания**

# 

Задание 1. (20% из 100)

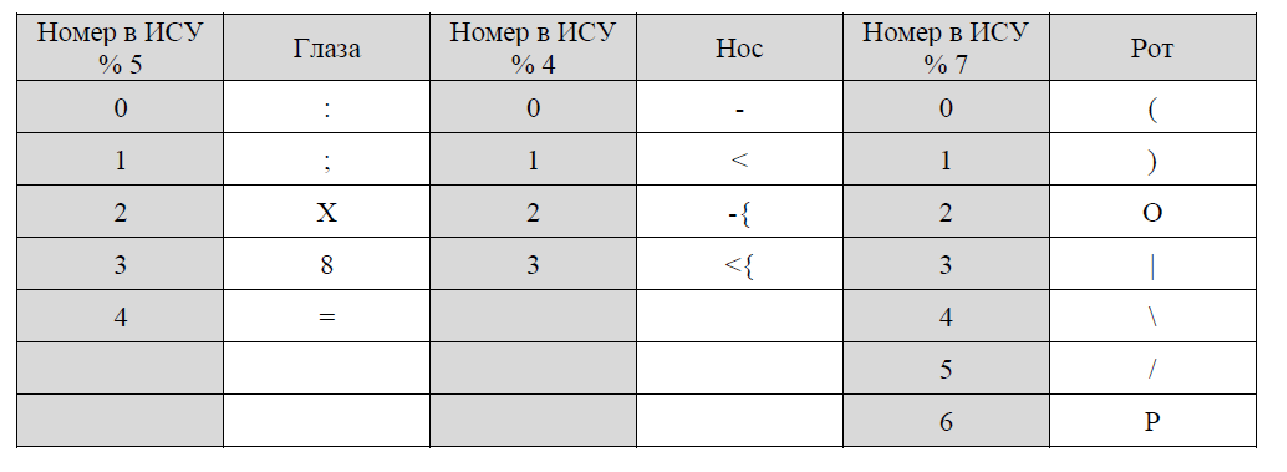
1) Реализуйте программный продукт на языке Python, используя регулярные выражения по варианту, представленному в таблице.

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.



Вариант 334:8<{\

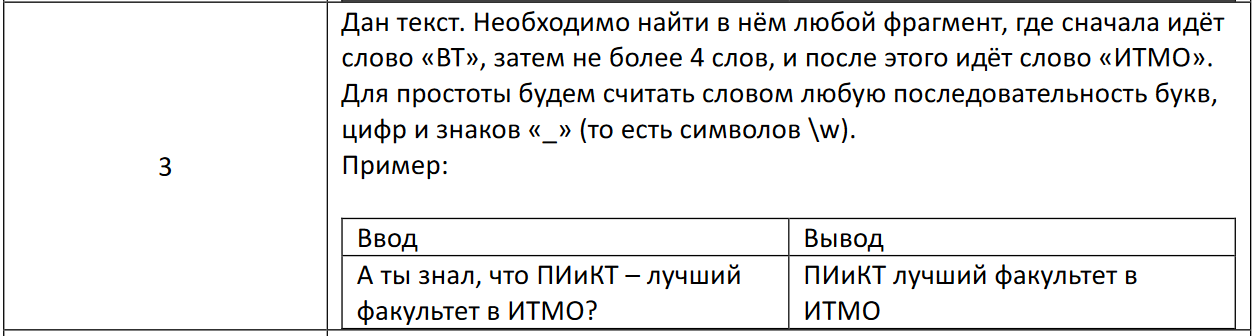
Задание 2. (40% из 100)

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



Вариант: 3

Задание 3.

1. Определить номер варианта как остаток деления номера в ИСУ на 36. В

случае, если в данный день недели нет занятий, то увеличить номер варианта

на восемь.

2. Изучить форму Бэкуса-Наура.

3. Изучить особенности протоколов и форматов обмена информацией между

системами: JSON, YAML, XML.

4. Понять устройство страницы с расписанием для своей группы:

5. Исходя из структуры расписания конкретного дня, сформировать файл с

расписанием в формате, указанном в задании в качестве исходного.

6. Обязательное задание: написать программу на языке Python 3.x, которая

бы осуществляла парсинг и конвертацию исходного файла в новый.

7. Нельзя использовать готовые библиотеки, в том числе регулярные

выражения в Python и библиотеки для загрузки XML-файлов.

8. Дополнительное задание №1

a) Найти готовые библиотеки, осуществляющие аналогичный парсинг и

конвертацию файлов.

b) Переписать исходный код, применив найденные библиотеки. Регулярные

выражения также нельзя использовать.

c) Сравнить полученные результаты и объяснить их сходство/различие.

9. Дополнительное задание №2

a) Переписать исходный код, добавив в него использование регулярных

выражений.

b) Сравнить полученные результаты и объяснить их сходство/различие.

27 JSON YAML Пятница

# **Выполнение работы:**

**Задание 1:**

**Вариант 334: 8<{\**

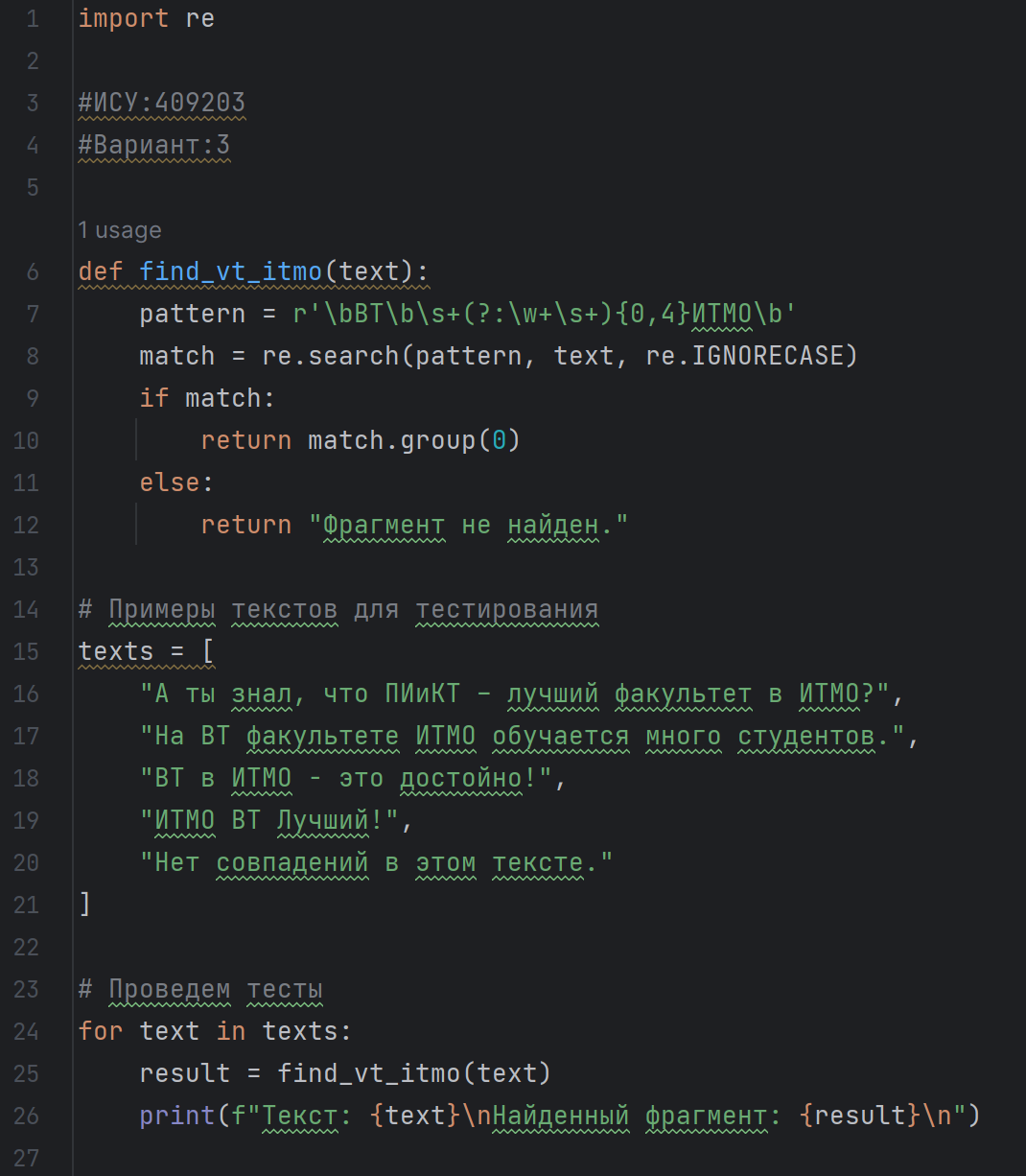


Код для решения задания 1



**Задание 2:**

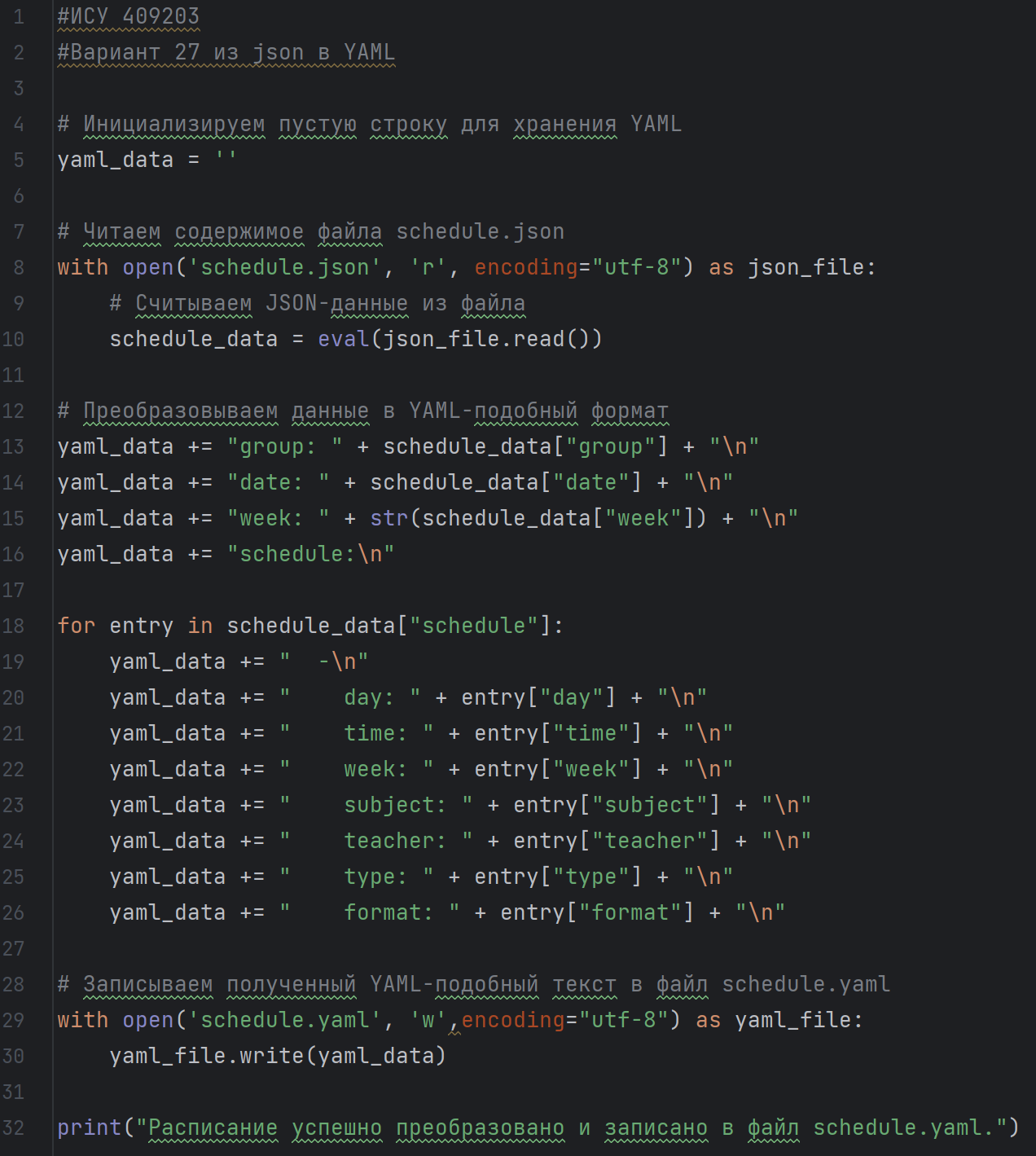
**Вариант 3:**



Код для решения задания 2

**Задание 3:**

**Вариант 27:**



Код для решения задания 3

**Задание 3.1:**



Код для решения задания 3.1

Сравнение:

#### **Преимущества второго кода (используя библиотеки json и yaml.dump):**

1. **Безопасность**: Второй код безопаснее, так как он использует библиотеку json для чтения JSON-данных. Использование eval (как в первом коде) может быть опасным, если в JSON-файле содержатся злонамеренные данные.
2. **Читаемость и удобство**: Второй код более читаем и легко понимаем, так как он использует словари и структурированный код для преобразования данных.
3. **Библиотеки**: Второй код использует библиотеки json и yaml, что делает его более стандартизированным и поддерживаемым.
4. **Настройка формата**: Второй код позволяет настроить формат YAML-файла с помощью параметров default\_flow\_style и allow\_unicode.

В целом, второй код предпочтительнее из-за своей безопасности и более явной структуры.

**Задание 3.2:**



Код для решения задания 3.2

Сравнение:

Оба кода в большинстве своей части идентичны и выполняют аналогичные действия, такие как чтение JSON-данных, преобразование их в YAML-подобный формат и запись в файл. Различия между ними минимальны и заключаются в использовании библиотеки re во втором коде для обработки данных.

Важно отметить, что использование re для обработки данных JSON и преобразования их в YAML не является оптимальным и менее надежным способом, поскольку JSON имеет специфический структурированный формат, который лучше обрабатывать с использованием библиотеки json, а преобразование в YAML лучше выполнять с использованием библиотеки yaml.

# Заключение

В процессе выполнения лабораторной работы я узнал что такое экранирование и как его использовать при решении задач. Также я узнал про парсинг и про JSON, YAML и XML, как конвертировать из одной системы в другую.

# Список литературы:

1. Лямин А.В., Череповская Е.Н. Объектно-ориентированное

программирование. Компьютерный практикум. – СПб: Университет ИТМО,

2017. – 143 с. – Режим доступа: https://books.ifmo.ru/file/pdf/2256.pdf.

2. https://ru.wikipedia.org/wiki/Форма\_Бэкуса\_—\_Наура.

3.«Пишем изящный парсер на Питоне»: https://habr.com/ru/post/309242/.