Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет Программной Инженерии и Компьютерной Техники

**Лабораторная работа по информатике №4**

Вариант: 14

Практик: Машина Екатерина Алексеевна

Выполнил: Состанов Тимур Айратович

Группа: P3114

Санкт-Петербург, 2022

**Оглавление**

[**Задание** 3](#_Toc120558961)

[**Ход работы** 4](#_Toc120558962)

[**Основное задание** 5](#_Toc120558963)

[**Дополнительное задание №1** 7](#_Toc120558964)

[**Дополнительное задание №2** 8](#_Toc120558965)

[**Дополнительное задание №3** 9](#_Toc120558966)

[**Вывод** 10](#_Toc120558967)

[**Список литературы** 10](#_Toc120558968)

# **Задание**

**Обязательное задание** (позволяет набрать до 65 процентов от максимального числа баллов БаРС за данную лабораторную):

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.

**Дополнительное задание №1**(позволяет набрать +10 процентов от максимального числа баллов БаРС за данную лабораторную):

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.

**Дополнительное задание №2** (позволяет набрать +10 процентов от максимального числа баллов БаРС за данную лабораторную):

a) Переписать исходный код, добавив в него использование регулярных выражений.

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

**Дополнительное задание №3** (позволяет набрать +10 процентов от максимального числа баллов БаРС за данную лабораторную):

a) Используя свою исходную программу из обязательного задания, программу из дополнительного задания №1 и программу из дополнительного задания №2, сравнить стократное время выполнения парсинга + конвертации в цикле.

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

**Дополнительное задание №4** (позволяет набрать +5 процентов от максимального числа баллов БаРС за данную лабораторную):

c) Переписать исходную программу, чтобы она осуществляла парсинг и конвертацию исходного файла в любой другой формат (кроме JSON, YAML, XML, HTML): PROTOBUF, TSV, CSV, WML и т.п.

d) Проанализировать полученные результаты, объяснить особенности использования формата.

# **Ход работы**

Исходный файл (timetable.json):

{  
 "timetable":{  
 "lang": "ru",  
 "group": "P3114",  
 "day": "wednesday",  
 "lessons":{  
 "lesson\_1":{  
 "time": "8:20-9:50",  
 "room": "Ауд. 2416 (бывш. 412)",  
 "place": "Кронверкский пр., д.49, лит.А",  
 "lesson": "Математика (продвинутый уровень)",  
 "type": "Практика",  
 "teacher": "Клепов Д."  
 },  
 "lesson\_2":{  
 "time": "10:00-11:30",  
 "room": "Ауд. 2416 (бывш. 412)",  
 "place": "Кронверкский пр., д.49, лит.А",  
 "lesson": "Математика (продвинутый уровень)",  
 "type": "Практика",  
 "teacher": "Клепов Д."  
 },  
 "lesson\_3":{  
 "time": "20:00-21:30",  
 "room": "Ауд. Вход, ул.Белорусская, д.6, лит.А",  
 "place": "ул.Белорусская, д.6, лит.А",  
 "lesson": "Беговой клуб",  
 "type": "Физкультура",  
 "teacher": "Фофанов Василий Сергеевич"  
 }  
 }  
 }  
}

(re\_timetable.json):

{"timetable":{"lang": "ru", "group": "P3114", "day": "wednesday", "lessons":{"lesson\_1":{"time": "8:20-9:50", "room": "Ауд. 2416 (бывш. 412)", "place": "Кронверкский пр., д.49, лит.А", "lesson": "Математика (продвинутый уровень)", "type": "Практика", "teacher": "Клепов Д."}, "lesson\_2":{"time": "10:00-11:30", "room": "Ауд. 2416 (бывш. 412)", "place": "Кронверкский пр., д.49, лит.А", "lesson": "Математика (продвинутый уровень)", "type": "Практика", "teacher": "Клепов Д."}, "lesson\_3":{"time": "20:00-21:30", "room": "Ауд. Вход, ул.Белорусская, д.6, лит.А", "place": "ул.Белорусская, д.6, лит.А", "lesson": "Беговой клуб", "type": "Физкультура", "teacher": "Фофанов Василий Сергеевич"}}}}

## **Основное задание**

import time  
  
all\_time = 0  
  
  
def get\_information(pattern\_information, information):  
 pattern\_keys = [i for i in pattern\_information.keys()]  
 while '"' in information:  
 delta = information[:information.index('"')]  
 information = information[(information.index('"') + 1):]  
 if delta in pattern\_keys:  
 information = information[(information.index('"') + 1):]  
 pattern\_information[delta] = information[:information.index('"')]  
 return pattern\_information  
  
  
def about\_timetable(information):  
 pattern\_information = {'lang': None,  
 'group': None,  
 'day': None,  
 'lessons': None  
 }  
 return get\_information(pattern\_information, information)  
  
  
def about\_lesson(information):  
 pattern\_information = {'time': None,  
 'room': None,  
 'place': None,  
 'lesson': None,  
 'type': None,  
 'teacher': None  
 }  
 return get\_information(pattern\_information, information)  
  
  
def start(json\_file, yaml\_file):  
 global all\_time  
 start\_time = time.perf\_counter()  
 flag\_timetable = False  
 nesting\_level = -1  
 input\_file = open(json\_file, 'r', encoding='utf-8')  
 output\_file = open(yaml\_file, 'w', encoding='utf-8')  
 for lines in input\_file.read().replace('{', '!{').replace('}', '}!').split('!'):  
 nesting\_level += lines.count('{')  
 if lines.count(':') == 1:  
 current\_line = lines[(lines.index('"') + 1):-2]  
 output\_file.write(' ' \* nesting\_level \* 2 + current\_line + ':\n')  
 elif lines.count(':') > 1:  
 if flag\_timetable:  
 for key, value in about\_lesson(lines).items():  
 output\_file.write(' ' \* nesting\_level \* 2  
 + "{0}: {1}".format(key, value) + '\n')  
 else:  
 for key, value in about\_timetable(lines + '""').items():  
 output\_file.write(' ' \* nesting\_level \* 2  
 + "{0}: {1}".format(key, value) + '\n')  
 flag\_timetable = True  
 nesting\_level -= lines.count('}')  
 input\_file.close()  
 output\_file.close()  
 all\_time += time.perf\_counter() - start\_time  
  
  
for \_ in range(100):  
 start('re\_timetable.json', 'timetable.yaml')  
print(all\_time)

Результат

timetable:  
 lang: ru  
 group: P3114  
 day: wednesday  
 lessons:  
 lesson\_1:  
 time: 8:20-9:50  
 room: Ауд. 2416 (бывш. 412)  
 place: Кронверкский пр. д.49 лит.А  
 lesson: Математика (продвинутый уровень)  
 type: Практика  
 teacher: Клепов Д.  
 lesson\_2:  
 time: 10:00-11:30  
 room: Ауд. 2416 (бывш. 412)  
 place: Кронверкский пр. д.49 лит.А  
 lesson: Математика (продвинутый уровень)  
 type: Практика  
 teacher: Клепов Д.  
 lesson\_3:  
 time: 20:00-21:30  
 room: Ауд. Вход ул.Белорусская д.6 лит.А  
 place: ул.Белорусская д.6 лит.А  
 lesson: Беговой клуб  
 type: Физкультура  
 teacher: Фофанов Василий Сергеевич

Я старался сделать основную программу максимально в общем виде, чтобы она могла работать и с файлами, записанными в одну строку. Также в этой программе я хотел больше работать непосредственно с данными, что, как мне кажется, у меня получилось.

## **Дополнительное задание №1**

import yaml  
import json  
import time  
  
start\_time = time.perf\_counter()  
for \_ in range(100):  
 with open('timetable.json', 'r', encoding='utf-8') as input\_file:  
 with open('timetable.yaml', 'w', encoding='utf-8') as output\_file:  
 parsed = str(yaml.dump(json.load(input\_file))).replace('\ ', '').\  
 replace(' ', '').replace('"', '')  
 output\_file.write(parsed.encode().decode('unicode-escape'))  
  
print(time.perf\_counter() - start\_time)

Результат

timetable:  
 day: wednesday  
 group: P3114  
 lang: ru  
 lessons:  
 lesson\_1:  
 lesson: Математика (продвинутый уровень)  
 place: Кронверкскийпр., д.49, лит.А  
 room: Ауд. 2416 (бывш. 412)  
 teacher: Клепов Д.  
 time: 8:20-9:50  
 type: Практика  
 lesson\_2:  
 lesson: Математика (продвинутый уровень)  
 place: Кронверкскийпр., д.49, лит.А  
 room: Ауд. 2416 (бывш. 412)  
 teacher: Клепов Д.  
 time: 10:00-11:30  
 type: Практика  
 lesson\_3:  
 lesson: Беговой клуб  
 place: ул.Белорусская, д.6, лит.А  
 room: Ауд. Вход, ул.Белорусская, д.6, лит.А  
 teacher: Фофанов Василий Сергеевич  
 time: 20:00-21:30  
 type: Физкультура

Полученный файл немного отличается от того, который я получил в основном задании: ключи (имена пар) расположены в алфавитном порядке.

В остальном полученные файлы совпадают

## **Дополнительное задание №2**

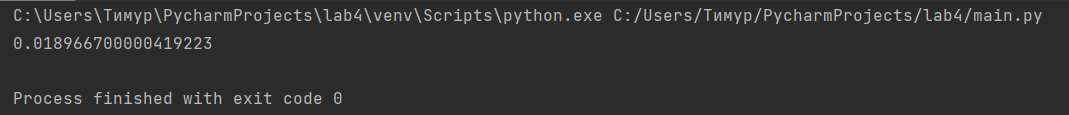
import re  
import time  
  
  
start\_time = time.perf\_counter()  
for \_ in range(100):  
 with open('timetable.json', 'r', encoding='utf-8') as json\_file:  
 with open('timetable.yaml', 'w', encoding='utf-8') as yaml\_file:  
 for current\_string in json\_file.read().split('\n'):  
 if len(re.findall(r'":', current\_string)) >= 1:  
 current\_string = re.sub(r'[{}"",]', '', current\_string)  
 yaml\_file.write(current\_string[2:] + '\n')  
print(time.perf\_counter() - start\_time)

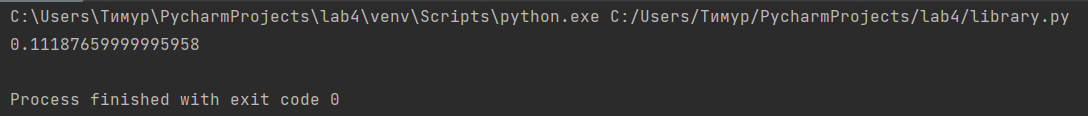
Результат

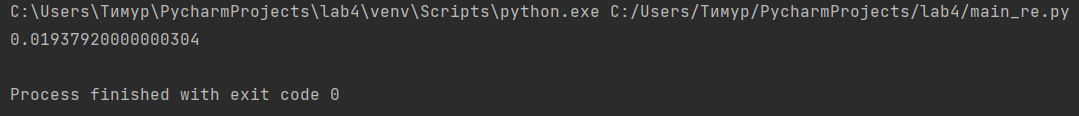
timetable:  
 lang: ru  
 group: P3114  
 day: wednesday  
 lessons:  
 lesson\_1:  
 time: 8:20-9:50  
 room: Ауд. 2416 (бывш. 412)  
 place: Кронверкский пр. д.49 лит.А  
 lesson: Математика (продвинутый уровень)  
 type: Практика  
 teacher: Клепов Д.  
 lesson\_2:  
 time: 10:00-11:30  
 room: Ауд. 2416 (бывш. 412)  
 place: Кронверкский пр. д.49 лит.А  
 lesson: Математика (продвинутый уровень)  
 type: Практика  
 teacher: Клепов Д.  
 lesson\_3:  
 time: 20:00-21:30  
 room: Ауд. Вход ул.Белорусская д.6 лит.А  
 place: ул.Белорусская д.6 лит.А  
 lesson: Беговой клуб  
 type: Физкультура  
 teacher: Фофанов Василий Сергеевич

Полученный файл полностью совпадает с файлом из основного задания

## **Дополнительное задание №3**

 *Рисунок 1 - результат работы основной программы*

 *Рисунок 2 – результат работы программы с библиотеками*

* Рисунок 3 – результат работы программы с регулярными выражениями*

Дольше всего работала программа с готовыми библиотеками, так как она сначала достает данные из json файла в читаемый python-ом формат (в многоуровневые словари), а потом из них производит полноценный парсинг. И еще немного костыльная обработка символов, записанных в кодировке utf-8, не добавляет скорости программе.

Код с регулярными выражениями работает немного дольше основного кода, так как сами регулярные выражения немного медленнее. Но разница незначительна из-за малых размеров читаемого файла и из-за того, что код с регулярными выражениями почти не обрабатывает файл, а просто меняет формат строк.

# **Вывод**

Во время выполнения лабораторной работы я узнал о языках разметки JSON, XML и YAML, научился с ними работать и переводить один тип данных в другой с помощью средств языка Python. Научился работать с библиотекой time для измерения времени и некоторыми библиотеками для автоматического парсинга между этими языками разметки.

К сожалению дополнительное задание №4 мне не поддалось, так как я испугался довольно размытого условия.

# **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>

1. «Пишем изящный парсер на Питоне»:

<https://habr.com/ru/post/309242/>