**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Факультет программной инженерии и компьютерной техники**

Системы искусственного интеллекта

Лабораторная работа № 4

Вариант № 13

**Выполнил студент**

Чан Дык Зюи

**Группа** № P3123

**Преподаватель**: Болдырева Елена Александровна

**г. Санкт-Петербург**

**2021**

**Оглавление**

[**Вариант: 336201 => 336201%36=33 Нет-то => 5 Нет-то =>13** 3](#_Toc88439087)

[**Задание:** 4](#_Toc88439088)

[**I.** **Порядок выполнения работы** 4](#_Toc88439089)

[**II.** **Решение** 5](#_Toc88439090)

[1) мой код: 5](#_Toc88439091)

[2) Дополнительное задание задание №1: 8](#_Toc88439092)

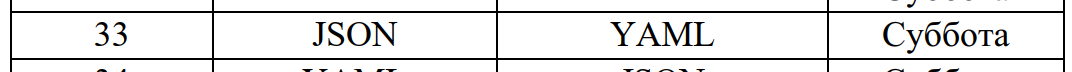
[3) Дополнительное задание задание №2 9](#_Toc88439093)

[4) Дополнительное задание задание №3 11](#_Toc88439094)

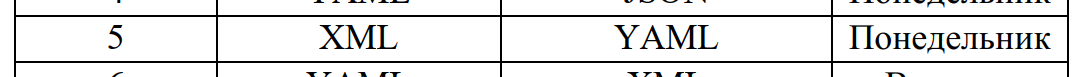
[5) Дополнительное задание задание №4 11](#_Toc88439095)

[**Вывод:** 12](#_Toc88439096)

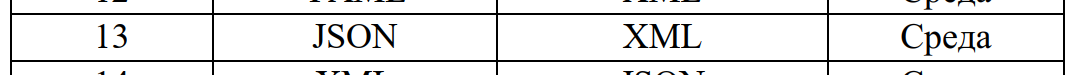
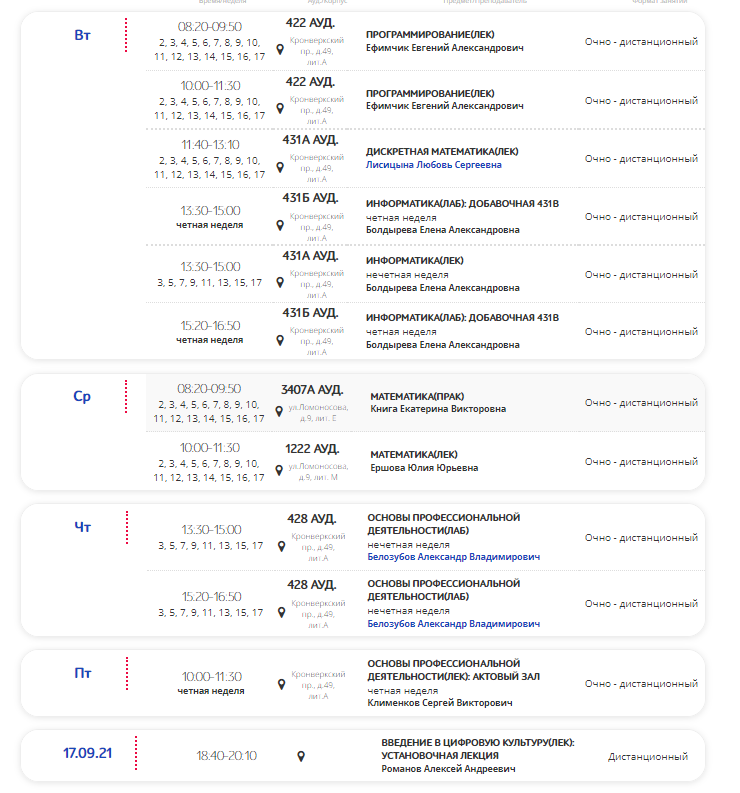
# **Вариант: 336201 => 336201%36=33 Нет-то => 5 Нет-то =>13**



Нет-то



нет -то



# **Задание:**

## **Порядок выполнения работы**

1. Определить номер варианта как остаток деления на 36 порядкового номера в списке группы в ISU. 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:  
https://itmo.ru/ru/schedule/0/P3123/schedule.htm  
5. Исходя из структуры расписания конкретного дня, сформировать файл с расписанием в формате, указанном в задании в качестве исходного.  
6. 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.  
8. Дополнительное задание 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.  
9. Дополнительное задание 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.  
10.Дополнительное задание 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 десятикратное время выполнения парсинга + конвертации в цикле.  
b) Проанализировать полученные результаты и объяснить их сходство/различие.  
11.Дополнительное задание задание №4 (позволяет набрать +5 процентов от максимального числа баллов БаРС за данную лабораторную.

a) Переписать исходную, чтобы она осуществляла парсинг и конвертацию исходного файла в любой другой формат (кроме JSON, YAML, XML, HTML): PROTOBUF, TSV, CSV, WML и т.п.  
b) Проанализировать полученные результаты, объяснить осоебнности использованного формата.  
12.Проверить, что все пункты задания выполнены и выполнены верно.  
13.Написать отчёт о проделанной работе.  
14.Подготовиться к устным вопросам на защите

## **Решение**

https://github.com/Tran-Duc-Duy/convertJsonToXML

input:

{

    "Wednesday": {

        "Lesson1": {

            "Time": "08:20-09:50",

            "Week": "all",

            "Room": "3407А",

            "Place": "ул.Ломоносова, д.9, лит. E",

            "Subject": "МАТЕМАТИКА(ПРАК)",

            "Teacher": "Книга Екатерина Викторовна",

            "Format": "Очно - дистанционный"

        },

        "Lesson2": {

            "Time": "10:00-11:30",

            "Week": "all",

            "Room": "1222",

            "Place": "ул.Ломоносова, д.9, лит. M",

            "Subject": "МАТЕМАТИКА(ЛЕК)",

            "Teacher": "Ершова Юлия Юрьевна",

            "Format": "Очно - дистанционный"

        }

    }

}

1. мой код:

import time

content =list() # content in file

tags    =list() # Tags

InFo    =list() # info of tag

types   =list() # type of tag

flag    =list() # Отмечает позицию тега, имеющего тип данных dict

lines =0

index =0

count =0

f =open("json\_s.json", "r",encoding='UTF-8')

start\_time = time.perf\_counter()

# read file and count the number of lines

txt = f.readline()

while txt:

    content.append(txt)

    lines += 1

    txt = f.readline()

for a in content: # поиск построчно

    b= a.find("\"") # найти позицию "

    c= a.find("\"",b+1) # найти позицию "

    if a[b+1:c].isalnum():

        tags.append(a[b+1:c])

        # найти положение {

        temp=a.find("{")

        # если не найден

        if temp==-1:

            types.append("str")

            d= a.find("\"",c+1) # найти позицию "

            e= a.find("\"",d+1) # найти позицию "

            InFo.append(a[d+1:e])

        # если найден

        else:

            types.append("dict")

            flag.append(len(types)-1)

            InFo.append("")

with open("xml\_sNoLib.xml", "w",encoding='UTF-8') as m:

    m.write("<?xml version=\"1.0\" ?>\n<all>\n")

    for a in range(1,lines-1-len(flag)):

        if(types[a-1]=="dict"):

            if(index!=0 and index!=-1):

                m.write("\t\t</"+tags[flag[count+1]]+">\n\t\t<"+tags[a-1]+" type=\""+types[a-1]+"\">\n")

                count+=1

                index= a-1

            #index =-1 => write second tag dict

            elif index==-1:

                m.write("\t\t<"+tags[a-1]+" type=\""+types[a-1]+"\">\n")

                index= a-1 # index =1

            #index =0 => write first tag dict

            else:

                m.write("\t<"+tags[a-1]+" type=\""+types[a-1]+"\">\n")

                index=-1 # index =-1

        else: # type = str

            m.write("\t\t\t<"+tags[a-1]+" type=\""+types[a-1]+"\"+>"+InFo[a-1]+"</"+tags[a-1]+">\n")

    tempC=count #count изменится

    for ind in range (tempC,len(flag)):

        if(ind ==len(flag)-1):

            m.write("\t</"+tags[flag[0]]+">\n")

        else:

            m.write("\t\t</"+tags[flag[count+1]]+">\n")

        count+=1

    m.write("</all>")

m.close()

f.close()

print(time.perf\_counter() - start\_time)

output:

<?xml version="1.0" ?>

<all>

    <Wednesday type="dict">

        <Lesson1 type="dict">

            <Time type="str">08:20-09:50</Time>

            <Week type="str">all</Week>

            <Room type="str">3407А</Room>

            <Place type="str">ул.Ломоносова, д.9, лит. E</Place>

            <Subject type="str">МАТЕМАТИКА(ПРАК)</Subject>

            <Teacher type="str">Книга Екатерина Викторовна</Teacher>

            <Format type="str">Очно - дистанционный</Format>

        </Lesson1>

        <Lesson2 type="dict">

            <Time type="str">10:00-11:30</Time>

            <Week type="str">all</Week>

            <Room type="str">1222</Room>

            <Place type="str">ул.Ломоносова, д.9, лит. M</Place>

            <Subject type="str">МАТЕМАТИКА(ЛЕК)</Subject>

            <Teacher type="str">Ершова Юлия Юрьевна</Teacher>

            <Format type="str">Очно - дистанционный</Format>

        </Lesson2>

    </Wednesday>

</all>

1. Дополнительное задание задание №1:

использовать готовые библиотеки

#import for csv

import pandas as pd

import csv

import time

#import for json

import json

from os import write

from json2xml import json2xml

from json2xml.utils import readfromurl, readfromstring, readfromjson

f =open("json\_s.json", "r",encoding='UTF-8')

start\_time = time.perf\_counter()

txt =f.read()

#to CSV

df = pd.read\_json (r'D:\DuyStudy\\FILE\_TinHocHocKy1\\TranDucDuy\_Lab4\_JsonToXML\\Json\_s.json')

df.to\_csv (r'D:\DuyStudy\\FILE\_TinHocHocKy1\\TranDucDuy\_Lab4\_JsonToXML\\csv\_s.csv', index = None)

#to XML

data = readfromstring(txt)

result = json2xml.Json2xml(data, wrapper="all", pretty=True).to\_xml()

#print(result) in file

m= open("xml\_sByLib.xml", "w",encoding='UTF-8')

m.write(result)

m.close()

f.close()

#count runtime

print(time.perf\_counter() - start\_time)

output:

<?xml version="1.0" ?>

<all>

    <Wednesday type="dict">

        <Lesson1 type="dict">

            <Time type="str">08:20-09:50</Time>

            <Week type="str">all</Week>

            <Room type="str">3407А</Room>

            <Place type="str">ул.Ломоносова, д.9, лит. E</Place>

            <Subject type="str">МАТЕМАТИКА(ПРАК)</Subject>

            <Teacher type="str">Книга Екатерина Викторовна</Teacher>

            <Format type="str">Очно - дистанционный</Format>

        </Lesson1>

        <Lesson2 type="dict">

            <Time type="str">10:00-11:30</Time>

            <Week type="str">all</Week>

            <Room type="str">1222</Room>

            <Place type="str">ул.Ломоносова, д.9, лит. M</Place>

            <Subject type="str">МАТЕМАТИКА(ЛЕК)</Subject>

            <Teacher type="str">Ершова Юлия Юрьевна</Teacher>

            <Format type="str">Очно - дистанционный</Format>

        </Lesson2>

    </Wednesday>

</all>

При использовании библиотеки программа имеет более высокую скорость обработки и большее разнообразие входных данных.

1. Дополнительное задание задание №2

"""

Сначал я использую регулярное выражение для поиска заголовков dict,

затем нахожу все заголовки,

затем нахожу заголовков str и,

наконец, нахожу информацию.

"""

import re

import time

f =open("json\_s.json", "r",encoding='UTF-8')

start\_time = time.perf\_counter()

txt = f.read()

#find dict

pattern = '\".+\"\:\s\{'

tagDict = re.findall(pattern,txt)

for i in range(len(tagDict)):

    tagDict[i]=tagDict[i][1:len(tagDict[i])-4]

#find str

pattern1 = '\".+\"\:\s[^{]'

tagStr = re.findall(pattern1,txt)

for i in range(len(tagStr)):

    tagStr[i] = tagStr[i][1:len(tagStr[i])-4]

#find info

pattern = '\:\s\".+\"'

tagInfo = re.findall(pattern,txt)

for i in range(len(tagInfo)):

    tagInfo[i] = tagInfo[i][3:len(tagInfo[i])-1]

count =0 #Переменная, в которой хранится значение, определяет порядок информации.

curDich=1 #Переменная, в которой хранится значение, - это порядок пары.

with open("./bt/xml\_s2.xml", "w",encoding='UTF-8') as m:

    m.write("<?xml version=\"1.0\" ?>\n<all>\n\t<"+tagDict[0]+" type=\"dict\">\n")

    """

    Поскольку есть только один день, заголовок первого словаря представляет день недели.

    """

    while (curDich < len(tagDict)):

        m.write("\t\t<"+tagDict[curDich]+" type=\"dict\">\n")

        while (count < len(tagStr)):

            if count >= curDich \*len(tagStr)//2:

                break

            m.write("\t\t\t<"+tagStr[count]+" type=\"str\">"+tagInfo[count]+"</"+tagStr[count]+">\n")

            count+=1

        m.write("\t\t</"+tagDict[curDich]+">\n")

        curDich+=1

    m.write("\t</"+tagDict[0]+">\n</all>")

m.close()

f.close()

print(time.perf\_counter() - start\_time)

output:

<?xml version="1.0" ?>

<all>

    <Wednesday type="dict">

        <Lesson1 type="dict">

            <Time type="str">08:20-09:50</Time>

            <Week type="str">all</Week>

            <Room type="str">3407А</Room>

            <Place type="str">ул.Ломоносова, д.9, лит. E</Place>

            <Subject type="str">МАТЕМАТИКА(ПРАК)</Subject>

            <Teacher type="str">Книга Екатерина Викторовна</Teacher>

            <Format type="str">Очно - дистанционный</Format>

        </Lesson1>

        <Lesson2 type="dict">

            <Time type="str">10:00-11:30</Time>

            <Week type="str">all</Week>

            <Room type="str">1222</Room>

            <Place type="str">ул.Ломоносова, д.9, лит. M</Place>

            <Subject type="str">МАТЕМАТИКА(ЛЕК)</Subject>

            <Teacher type="str">Ершова Юлия Юрьевна</Teacher>

            <Format type="str">Очно - дистанционный</Format>

        </Lesson2>

    </Wednesday>

</all>

1. Дополнительное задание задание №3

|  |  |  |
| --- | --- | --- |
| Не использовать готовые библиотеки | использовать готовые библиотеки | использование регулярных выражений. |
| 0.0010481999488547444 | 0.02343010006006807 | 0.0018885000608861446 |
| 0.0009147000964730978 | 0.022027999977581203 | 0.0016132999444380403 |
| 0.001000800053589046 | 0.0218297999817878 | 0.0014593000523746014 |
| 0.0008100999984890223 | 0.02369539998471737 | 0.0016660999972373247 |
| 0.0017061999533325434 | 0.016745199915021658 | 0.0011427999706938863 |

Программа, использующая доступную библиотеку, имеет самое продолжительное время выполнения.

Программа, использующая регулярные выражения, имеет второе по продолжительности время выполнения

Самое быстрое время выполнения - обычная программа с обычным поиском.

Я думаю, что заявление re.replace не может ускорить меня.

Переписывание функции для преобразования занимает много времени, а природа каждого файла разная, поэтому использовать функцию удобнее. Но если при написании программы нужно сильно оптимизировать, я перепишу функцию, чтобы она была максимально приближена к типу данных, используемому в файле.

1. Дополнительное задание задание №4

import pandas as pd

import csv

import json

df = pd.read\_json (r'D:\DuyStudy\\FILE\_TinHocHocKy1\\lab4\\Json\_s.json')

df.to\_csv (r'D:\DuyStudy\\FILE\_TinHocHocKy1\\lab4\\bt\\csv\_s.csv', index = None)

output:

Wednesday

"{'Time': '08:20-09:50', 'Week': 'all', 'Room': '3407А', 'Place': 'ул.Ломоносова, д.9, лит. E', 'Subject': 'МАТЕМАТИКА(ПРАК)', 'Teacher': 'Книга Екатерина Викторовна', 'Format': 'Очно - дистанционный'}"

"{'Time': '10:00-11:30', 'Week': 'all', 'Room': '1222', 'Place': 'ул.Ломоносова, д.9, лит. M', 'Subject': 'МАТЕМАТИКА(ЛЕК)', 'Teacher': 'Ершова Юлия Юрьевна', 'Format': 'Очно - дистанционный'}"

* файлы с расширением .tsv и .csv имеют структуру столбцов, аналогичную Excel .xlsx, и оба они разработаны на основе формата .xml.

# **Вывод:**

Понимать и знать, как работать с типами файлов. Понять способы хранения JSON, XML, YAML, CSV, HTML ... Преобразование информации из одного типа файла в другой.

**Список литературы**

<https://pypi.org/project/json2xml/>

<https://www.geeksforgeeks.org/python-json-to-xml/?ref=lbp>