

МИНИСТЕРСТВО НАУКИ И ОБРАЗОВАНИЯ РФ

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

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

ЛАБОРАТОРНАЯ РАБОТА №4

«Исследование протоколов, форматов обмена
информацией и языков разметки документов»

по дисциплине
ИНФОРМАТИКА

Вариант №-16

Выполнила:

Студент группы Р3132

Чмурова Мария

Владиславовна

Проверил:

Белозубов

Александр Владимирович

г. Санкт-Петербург
2022 год

Оглавление

Задание.....	3
Основные этапы вычислений	5
Основное задание №1	5
Дополнительное задание №1	7
Дополнительное задание №2	8
Заключение	11
Список используемой литературы	12

Задание

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 языков разметки/форматов JSON, YAML, XML.
4. Понять устройство страницы с расписанием для своей группы:
<http://itmo.ru/ru/schedule/0/P3110/schedule.htm>
5. Исходя из структуры расписания конкретного дня, сформировать файл с расписанием в формате, указанном в задании в качестве исходного. При этом необходимо, чтобы в выбранном дне было не менее двух занятий (можно использовать своё персональное). В случае, если в данный день недели нет таких занятий, то увеличить номер варианта ещё на восемь.
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 №1 (позволяет набрать +10 процентов от максимального числа баллов БаРС за данную лабораторную).
 - Найти готовые библиотеки, осуществляющие аналогичный парсинг и конвертацию файлов.
 - Переписать исходный код, применив найденные библиотеки. Регулярные выражения также нельзя использовать.
 - Сравнить полученные результаты и объяснить их сходство/различие.
9. Дополнительное задание №2 (позволяет набрать +10 процентов от максимального числа баллов БаРС за данную лабораторную).

- Переписать исходный код, добавив в него использование регулярных выражений.
 - Сравнить полученные результаты и объяснить их сходство/различие.
10. Дополнительное задание №3 (позволяет набрать +10 процентов от максимального числа баллов БаРС за данную лабораторную).
- 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.
 - Проанализировать полученные результаты и объяснить их сходство/различие.
11. Дополнительное задание №4 (позволяет набрать +5 процентов от максимального числа баллов БаРС за данную лабораторную).
- с) 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.
12. Проверить, что все пункты задания выполнены и выполнены верно.
13. Написать отчёт о проделанной работе.
14. Подготовиться к устным вопросам на защите.

Основные этапы вычислений

Основное задание №1

Расписание на среду: schedule_tuesday.yaml

```
wednesday:
- lesson1:
  name: 'Информатика (ЛАБ) '
  start_time: '8:20'
  end_time: '9:50'
  place: 'Кронверкский пр., д.49, лит.А'
  teachers_name: 'Белозубов Александр Владимирович'
  classroom: '305'
  format: 'Очно-Дистанционный'
  odd_week: 'true'
- lesson2:
  name: 'Информатика (ЛАБ) '
  start_time: '10:00'
  end_time: '11:30'
  place: 'Кронверкский пр., д.49, лит.А'
  teachers_name: 'Белозубов Александр Владимирович'
  classroom: '305'
  format: 'Очно-Дистанционный'
  odd_week: 'true'
```

Основное задание №1: main.py

```
import time

yaml = open("schedule_tuesday.yaml", "r") #открытие yaml-файла для записи (r)
xml = open("schedule_tuesday.xml", "w") #открытие xml-файла для чтения (w)
xml.write('<?xml version="1.0" encoding="UTF-8"?>\n') #заполняем XML пролог в xml-файл и добавляем переход на следующую строку
indent = '  '
mas = []
nes = []
flag = False

start_time = time.time()

for line in yaml:
  a = line.split(": ", 1)
  nesting = line.count(indent)
  ex = a[0].lstrip()[0]
  if ex == "-":
    nesting += 1
    a[0] = a[0].lstrip()
    a[0] = a[0][2: ]

    if ex == "-" and flag == True:
```

```

        xml.write((indent*(len(mas)-1)) + "</" + mas[-1] + ">" +
"\n")

    if nesting not in nes:
        nes.append(nesting)
        nes.sort()
    elif nesting < nes[-1] and ex != '-':
        for i in range(nes[-1] - nesting):
            xml.write((indent * (len(mas)-1)) + "</" + mas[-1] +
">" + "\n")
            del mas[-1]
            del nes[-1]

    if len(a) == 1:
        xml.write((indent*nesting) + "<" + a[0][: -2].rstrip() +
">" + "\n")
        if flag == True and ex == "-":
            del mas[-1]
            mas.append(a[0][: -2].rstrip())

    if len(a) == 2:
        a[1] = a[1][1: -2]
        a[0] = a[0].rstrip()

        xml.write((indent * nesting) + "<" + a[0] + ">" + a[1] +
"</" + a[0] + ">" + '\n')

    if ex == "-" and flag == False:
        flag = True

if mas != []:
    for i in range(len(mas)):
        xml.write((indent * (len(mas)-1)) + "</" + mas[-1] + ">"
+ "\n")
        del mas[-1]

print((time.time()-start_time) * 100)

```

Результат работы программы для файла schedule_tuesday.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<wednesday>
  <lesson1>
    <name>Информатика (ЛАБ) </name>
    <start_time>8:20</start_time>
    <end_time>9:50</end_time>
    <place>Кронверкский пр., д.49, лит.А</place>
    <teachers_name>Белозубов Александр
Владимирович</teachers_name>
    <classroom>305</classroom>
    <format>Очно-Дистанционный</format>
    <odd_week>true</odd_week>
  </lesson1>

```

```

<lesson2>
  <name>Информатика (ЛАБ) </name>
  <start_time>10:00</start_time>
  <end_time>11:30</end_time>
  <place>Кронверкский пр., д.49, лит.А</place>
  <teachers_name>Белозубов Александр
Владимирович</teachers_name>
  <classroom>305</classroom>
  <format>Очно-Дистанционный</format>
  <odd_week>tru</odd_week>
</lesson2>
</wednesday>

```

Дополнительное задание №1

Дополнительное задание №1: additional_task_1

```

import xmlplain
import time

start_time = time.time()

with open("schedule_tuesday.yaml", "r", encoding='utf-8') as inf:
    root = xmlplain.obj_from_yaml(inf)

with open("schedule_tuesday.xml", "w", encoding='utf-8') as outf:
    xmlplain.xml_from_obj(root, outf, pretty=True)

print((time.time()-start_time) * 100)

```

Результат работы программы для файла schedule_tuesday.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<wednesday>
  <lesson1>
    <name>Информатика (ЛАБ) </name>
    <start_time>8:20</start_time>
    <end_time>9:50</end_time>
    <place>Кронверкский пр., д.49, лит.А</place>
    <teachers_name>Белозубов Александр
Владимирович</teachers_name>
    <classroom>305</classroom>
    <format>Очно-Дистанционный</format>
    <odd_week>true</odd_week>
  </lesson1>
  <lesson2>
    <name>Информатика (ЛАБ) </name>
    <start_time>10:00</start_time>
    <end_time>11:30</end_time>
    <place>Кронверкский пр., д.49, лит.А</place>
    <teachers_name>Белозубов Александр
Владимирович</teachers_name>
    <classroom>305</classroom>

```

```

    <format>Очно-Дистанционный</format>
    <odd_week>tru</odd_week>
</lesson2>
</wednesday>

```

Модуль `xmlplain` представляет собой набор служебных функций для разбора автономных входных данных XML в простые типы `list` / `dict` / `string` и передачи в / чтения из форматов XML или YAML.

Данный модуль обеспечивает:

- Простые объекты, выгруженные в/перезагруженные из XML
- Сохранение полуструктурированных XML-документов (дубликатов тегов, смешанного текста и тегов) при вводе
- Управление редактируемой человеком формой через YAML мост

Дополнительное задание №2

Дополнительное задание №2: `additiona_task_2.py`

```

import re
import time

yaml = open("schedule_tuesday.yaml", "r") #открытие yaml-файла
для записи (r)
xml = open("schedule_tuesday.xml", "w") #открытие xml-файла для
чтения (w)
xml.write('<?xml version="1.0" encoding="UTF-8"?>\n')
indent = ' '
mas = []
nes = []
flag = False

start_time = time.time()

for line in yaml:
    reg = re.split(r':', line, maxsplit=1)
    nesting = line.count(indent)
    ex = re.findall(r'[\S]', reg[0])[0]

    if ex == "-":
        nesting += 1
        part = "".join(re.findall(r"\w+", reg[0]))

    if ex == "-" and flag == True:
        xml.write((indent*(len(mas)-1)) + "</" + mas[-1] + ">" +
"\n")

    if nesting not in nes:

```



```

        nes.append(nesting)
        nes.sort()
        elif nesting < nes[-1] and ex != '-':
            for i in range(nes[-1] - nesting):
                xml.write((indent * (len(mas)-1)) + "</" + mas[-1] +
">" + "\n")
                del mas[-1]
                del nes[-1]

            if reg[1] != "\n":
                first_part = "".join(re.findall(r"[-\S]+\w", reg[0]))
                second_part = "".join(re.findall(r"'([^\']*)'", reg[1]))
                xml.write((indent * nesting) + "<" + first_part + ">" +
second_part + "</" + first_part + ">" + '\n')
            else:
                part = "".join(re.findall(r"\w+", reg[0]))
                xml.write((indent * nesting) + "<" + part + ">" + "\n")
                if flag == True and ex == "-":
                    del mas[-1]
                    mas.append(part)

            if ex == "-" and flag == False:
                flag = True

if mas != []:
    for i in range(len(mas)):
        xml.write((indent * (len(mas)-1)) + "</" + mas[-1] + ">"
+ "\n")
        del mas[-1]

print((time.time() - start_time) * 100)

```

Результат работы программы для файла schedule_tuesday.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<wednesday>
  <lesson1>
    <name>Информатика (ЛАБ) </name>
    <start_time>8:20</start_time>
    <end_time>9:50</end_time>
    <place>Кронверкский пр., д.49, лит.А</place>
    <teachers_name>Белозубов Александр
Владимирович</teachers_name>
    <classroom>305</classroom>
    <format>Очно-Дистанционный</format>
    <odd_week>true</odd_week>
  </lesson1>
  <lesson2>
    <name>Информатика (ЛАБ) </name>
    <start_time>10:00</start_time>
    <end_time>11:30</end_time>
    <place>Кронверкский пр., д.49, лит.А</place>

```

```
<teachers_name>Белозубов Александр  
Владимирович</teachers_name>  
<classroom>305</classroom>  
<format>Очно-Дистанционный</format>  
<odd_week>tru</odd_week>  
</lesson2>  
</wednesday>
```

Заключение

В ходе данной лабораторной работы, я узнала как писать парсер для перевода уaml-файлов в xml-файлы, а также как делать это с использованием библиотеки и регулярных выражений.

Список используемой литературы

Балакшин П.В., Соснин В.В., Машина Е.А. Информатика. – СПб: Университет ИТМО, 2020. – 122 с.