

**Министерство науки и высшего образования Российской Федерации**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧЕРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
ИТМО»**

**Факультет ПИиКТ**

**Дисциплина: Информатика**

**Лабораторная работа №4  
Исследование протоколов, форматов обмена  
информацией и языков разметки документов**

**Вариант 30**

**Выполнил: Михайлов Петр Сергеевич**

**Группа: Р3111**

**Преподаватель: доцент, кандидат технических наук**

**Малышева Татьяна Алексеевна**

**Санкт-Петербург 2024г.**

## Содержание

Задание.....	3
Основные этапы вычисления .....	5
Исходное расписание и json-файл .....	5
Задание 1 (Основное задание) .....	6
Задание 2 (Дополнительное №1).....	6
Задание 3 (Дополнительное №2).....	6
Задание 4 (Дополнительное №3).....	6
Задание 5 (Дополнительное №4).....	7
Задание 6 (Дополнительное №5).....	7
Заключение.....	8
Литература .....	9

## Задание

1. Определить номер варианта как остаток деления на 36 последних двух цифр своего идентификационного номера в ISU: например, **466730** % 36 = 30. В случае, если в оба указанных дня недели нет занятий, то увеличить номер варианта на восемь. В случае, если занятий нет и в новом наборе дней, то продолжать увеличивать на восемь.
2. Изучить форму Бэкуса-Наура.
3. Изучить основные принципы организации формальных грамматик.
4. Изучить особенности языков разметки/форматов JSON, YAML, XML.
5. Понять устройство страницы с расписанием на примере расписания лектора: [https://itmo.ru/ru/schedule/3/125598/raspisanie\\_zanyatiy.htm](https://itmo.ru/ru/schedule/3/125598/raspisanie_zanyatiy.htm)
6. Исходя из структуры расписания конкретного дня, сформировать файл с расписанием в формате, указанном в задании в качестве исходного. При этом необходимо, чтобы хотя бы в одной из выбранных дней 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.
7. **Обязательное задание** (позволяет набрать до 45 процентов от максимального числа баллов БаРС за данную лабораторную): написать программу на языке Python 3.x или любом другом, которая бы осуществляла парсинг и конвертацию исходного файла в новый путём простой замены метасимволов исходного формата на метасимволы результирующего формата.
8. Нельзя использовать готовые библиотеки, в том числе регулярные выражения в Python и библиотеки для загрузки XML-файлов.
9. **Дополнительное задание №1** (позволяет набрать +10 процентов от максимального числа баллов БаРС за данную лабораторную).
  - а) Найти готовые библиотеки, осуществляющие аналогичный парсинг и конвертацию файлов.
  - б) Переписать исходный код, применив найденные библиотеки. Регулярные выражения также нельзя использовать.
  - в) Сравнить полученные результаты и объяснить их сходство/различие. Объяснение должно быть отражено в отчёте.
10. **Дополнительное задание №2** (позволяет набрать +10 процентов от максимального числа баллов БаРС за данную лабораторную).
  - а) Переписать исходный код, добавив в него использование регулярных выражений.
  - б) Сравнить полученные результаты и объяснить их сходство/различие. Объяснение должно быть отражено в отчёте.
11. **Дополнительное задание №3** (позволяет набрать +25 процентов от максимального числа баллов БаРС за данную лабораторную).
  - а) Переписать исходный код таким образом, чтобы для решения задачи использовались формальные грамматики. То есть ваш код должен уметь осуществлять парсинг и конвертацию любых данных, представленных в исходном формате, в данные, представленные в результирующем формате: как с готовыми библиотеками из дополнительного задания №1.

- b) Проверку осуществить как минимум для расписания с двумя учебными днями по два занятия в каждом.
  - c) Сравнить полученные результаты и объяснить их сходство/различие. Объяснение должно быть отражено в отчёте.
12. **Дополнительное задание №4** (позволяет набрать +5 процентов от максимального числа баллов БаРС за данную лабораторную).
- a) Используя свою исходную программу из обязательного задания и программы из дополнительных заданий, сравнить стократное время выполнения парсинга + конвертации в цикле.
  - b) Проанализировать полученные результаты и объяснить их сходство/различие. Объяснение должно быть отражено в отчёте.
13. **Дополнительное задание №5** (позволяет набрать +5 процентов от максимального числа баллов БаРС за данную лабораторную).
- a) 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.
  - b) Проанализировать полученные результаты, объяснить особенности использования формата. Объяснение должно быть отражено в отчёте.
14. Проверить, что все пункты задания выполнены и выполнены верно.
15. Написать отчёт о проделанной работе.
16. Подготовиться к устным вопросам на защите.

## Основные этапы вычисления

### Исходное расписание и json-файл

С6	13:30-15:00 2, 4, 6, 8, 10, 12, 14, 16	Информ 1.2 (Р3107)	1415 АУД. Кронверкский пр., д.49, лит.А	ИНФОРМАТИКА (ЛАБ)	Очно - дистанционный
	13:30-15:00 1, 3, 5, 7, 9, 11, 13, 15	Информ 1.1 (Р3106)	Кронверкский пр., д.49, лит.А	ИНФОРМАТИКА (ЛАБ)	Очно - дистанционный
	15:20-16:50 2, 4, 6, 8, 10, 12, 14, 16	Информ 1.2 (Р3107)	Кронверкский пр., д.49, лит.А	ИНФОРМАТИКА (ЛАБ)	Очно - дистанционный
	15:20-16:50 1, 3, 5, 7, 9, 11, 13, 15	Информ 1.1 (Р3106)	Кронверкский пр., д.49, лит.А	ИНФОРМАТИКА (ЛАБ)	Очно - дистанционный

Рис. 1 Исходное расписание – расписание П. В. Балакшина

Начало из примерного файла с расписанием в формате JSON:

```
{
  "current_day": 6,
  "current_week": 9,
  "current_time": "19:25",
  "schedule": [
    {
      "id": 1000,
      "title": "Информатика",
      "group": "Информ 1.2 (Р3107)",
      "teacher": {
        "id": 125598,
        "name": "Балакшин Павел Валерьевич"
      },
      "class_type": {
        "id": 2,
        "name": "Лабораторная"
      },
      "class_format": {
        "id": 2,
        "name": "Очно - дистанционный"
      },
      "date": {
        "day": 6,
        "weeks": [ 2, 4, 6, 8, 10, 12, 14, 16 ]
      },
      "time": {
        "class_number": 4,
        "from": "13:30",
        "to": "15:00"
      },
      "classroom": {
        "id": 1415,
        "name": "1415 ауд."
      },
      "campus": {
        "id": 0,
        "adress": "Кронверкский пр., д.49, лит.А"
      }
    },
    ...
  ]
}
```

## Задание 1 (Основное задание)

Исходный файл json: <https://clck.ru/3EPyMg>

Исходный код программы: <https://clck.ru/3EPyKx>

Результат вывода программы: <https://clck.ru/3EPyNX>

## Задание 2 (Дополнительное №1)

Исходный код программы: <https://clck.ru/3EPyKF>

Результат вывода программы: <https://clck.ru/3EPyPi>

Готовые библиотеки: стандартная библиотека Python json для парсинга JSON и dicttoxml для дампа словаря в файл xml.

Файл результата не отличается от результата обязательного задания, кроме как тем, что при использовании библиотеки элементы списков заключены в парный тэг “<item>”, когда при использовании самописной реализации используется “<[key]\_elem>”. Код программы стал значительно проще – теперь он состоит из одной функции и чтения файлов.

## Задание 3 (Дополнительное №2)

Исходный код программы: <https://clck.ru/3EPyHh>

Результат вывода программы: <https://clck.ru/3EPyRj>

Файл результата не отличается от результата обязательного задания. Единственное изменение в коде программы – парсинг чисел и строк заменены на регулярные выражения вместо циклов.

## Задание 4 (Дополнительное №3)

Исходный код программы: <https://clck.ru/3EPyEE>

Результат вывода программы: <https://clck.ru/3EPySj>

Изначальный код был написан с использованием формальных грамматик. Поэтому я смог импортировать нужные мне функции из Задания 1.

## Задание 5 (Дополнительное №4)

Исходный код программы: <https://clck.ru/3EPyCP>

Результат вывода программы:

```
Own Parser: 2.8399758338928223 seconds
Own Parser with Regex: 3.220170259475708 seconds
Python JSON Lib + dicttoxml: 2.139697790145874 seconds
Press any key to continue . . . |
```

Рис. 2 Сравнение времени выполнения трех программ

Результат оказался предсказуемым. Встроенная в Python библиотека JSON в сочетании с dicttoxml выполняет перевод наиболее быстрым образом. Несмотря на то, что хоть и таким способом выполняется множество проверок внутри библиотек, для того чтобы максимально точно соответствовать формату xml и не допускать ошибок, самописный парсер все равно не может сравниться в производительности, хотя его принцип работы основан на том, чтобы «обернуть» значение в тэг ключа с форматированными пробелами. Регулярные выражения работают дольше всего, так как сначала программа компилирует их (re.compile), а после идет проверять весь текст, что медленнее, чем простая итерация проверки по строке.

## Задание 6 (Дополнительное №5)

Исходный код программы: <https://clck.ru/3EPy9J>

Результат вывода программы: <https://clck.ru/3EPy7R>

TSV (англ. tab separated values — значения, разделённые табуляцией) — текстовый формат для представления таблиц баз данных. Каждая запись в таблице — это строка текстового файла. Каждое поле записи отделяется от других с помощью символа табуляции, точнее горизонтальной табуляции.

1	id	title	group	teacher/id	teacher/name	class_type/id	class_type/name	class_format/id	class_format/name	date/day	date/weeks/0	da
2	1000	Информатика	Информ 1.2 (Р3107)	125598	Балакшин Павел Валерьевич	2	Лабораторная	2	Очно - дистанционный	6	2	4
3	1001	Информатика	Информ 1.1 (Р3106)	125598	Балакшин Павел Валерьевич	2	Лабораторная	2	Очно - дистанционный	6	1	3
4	1002	Информатика	Информ 1.2 (Р3107)	125598	Балакшин Павел Валерьевич	2	Лабораторная	2	Очно - дистанционный	6	2	4
5	1003	Информатика	Информ 1.1 (Р3106)	125598	Балакшин Павел Валерьевич	2	Лабораторная	2	Очно - дистанционный	6	1	3

Рис. 3 Пример полученный таблицы после конвертации

## Заключение

В результате выполнения лабораторной работы, я познакомился с разными форматами файлов. Узнал о существовании формальных грамматик и БНФ. Написал парсеры, которые конвертируют json-файл в формат xml и tsv.



## Литература

- 1) Форма Бэкуса – Наура [Электронный ресурс]: Википедия. Свободная энциклопедия. – Режим доступа: <https://clck.ru/3EPqoV> (дата обращения: 04.11.2024).
- 2) Информатика: методическое пособие, 2020 [Электронный ресурс]: сост. и ред. Балакшин П.В., Соснин В.В., Машина Е.А. – Режим доступа: [https://vk.com/doc-31201840\\_566998093](https://vk.com/doc-31201840_566998093) (дата обращения: 04.11.2024).
- 3) Информатика: лабораторные работы и тесты, 2019 [Электронный ресурс]: Балакшин П.В., Соснин В.В., Калинин И.В., Малышева Т.А., Раков С.В., Рущенко Н.Г., Дергачев А.М. Редакционно-издательский отдел Университета ИТМО. – Режим доступа: <https://books.ifmo.ru/file/pdf/2464.pdf> (дата обращения: 04.11.2024).