МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Национальный исследовательский университет ИТМО»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И  
КОМПЬЮТЕРНОЙ ТЕХНИКИ

**ЛАБОРАТОРНАЯ РАБОТА №2**

по дисциплине

«ИНФОРМАТИКА»

Синтез помехоустойчивого кода

Вариант №80

***Выполнил:***Студент группы P3107  
Шишкин Артём Владимирович

***Проверил:***Балакшин Павел Валерьевич

кандидат технических наук, доцент факультета ПИиКТ

Санкт-Петербург, 2024

**Содержание**

[Задание 3](#_Toc183267278)

[Основные этапы вычисления 5](#_Toc183267279)

[Заключение 8](#_Toc183267280)

[Список литературы 9](#_Toc183267281)

# **Задание**

1. Определить номер варианта как остаток деления на 36 последних двух цифр своего идентификационного номера в ISU: например, 125598 / 36 = 26. В случае, если в оба указнных 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>

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).

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. Объяснение должно быть отражено в отчёте.

10.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. 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.

с) 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. 2

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**

1. Вариант №468100 % 36 = 0

JSON to XML, Понедельник, Четверг

1. Исходное расписание в формате JSON:

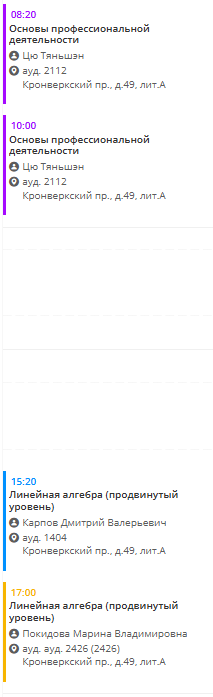


Рисунок 1 (Расписание Понедельник)

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 2 (Расписание Четверг)

{  
 "even\_week": true,  
 "group\_name": "P3107",  
 "schedule": [  
 {  
 "monday": [  
 {  
 "title": "Основы профессиональной деятельности",  
 "type": "Лабораторная",  
 "campus": "Кронверкский пр., д.49, лит.А",  
 "time": {  
 "from\_time": "08:20",  
 "to\_time": "09:50"  
 },  
 "auditory": "ауд. 2112",  
 "teacher": "Цю Тяньшэн"  
 },  
 {  
 "title": "Основы профессиональной деятельности",  
 "type": "Лабораторная",  
 "campus": "Кронверкский пр., д.49, лит.А",  
 "time": {  
 "from\_time": "10:00",  
 "to\_time": "11:30"  
 },  
 "auditory": "ауд. 2112",  
 "teacher": "Цю Тяньшэн"  
 },  
 {  
 "title": "Линейная алгебра (продвинутый уровень)",  
 "type": "Лекция",  
 "campus": "Кронверкский пр., д.49, лит.А",  
 "time": {  
 "from\_time": "15:20",  
 "to\_time": "16:50"  
 },  
 "auditory": "ауд. 1404",  
 "teacher": "Карпов Дмитрий Валерьевич"  
 },  
 {  
 "title": "Линейная алгебра (продвинутый уровень)",  
 "type": "Практика",  
 "campus": "Кронверкский пр., д.49, лит.А",  
 "time": {  
 "from\_time": "17:00",  
 "to\_time": "18:30"  
 },  
 "auditory": "ауд. 2426(2426)",  
 "teacher": "Покидова Марина Владимировна"  
 }  
 ]  
 },  
 {  
 "thursday": [  
 {  
 "title": "История русской культуры в контексте мировой культуры",  
 "type": "Лекция",  
 "campus": "Кронверкский пр., д.49, лит.А",  
 "time": {  
 "from\_time": "11:40",  
 "to\_time": "13:10"  
 },  
 "auditory": "ауд. Lemon Classroom (1419)",  
 "teacher": "Жиркова Галина Петровна"  
 },  
 {  
 "title": "История русской культуры в контексте мировой культуры",  
 "type": "Практика",  
 "campus": "Кронверкский пр., д.49, лит.А",  
 "time": {  
 "from\_time": "13:30",  
 "to\_time": "15:00"  
 },  
 "auditory": "ауд. 2412",  
 "teacher": "Щеглова Иванна Павловна"  
 }  
 ]  
 }  
 ]  
}

1. Основное задание

<https://github.com/ShobakaBarobaka/ITMO/blob/main/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0%D0%9B%D0%B0%D0%B1%D1%8B/lab4/jsonToXMLmanually.py>

1. Дополнительное задание №1

<https://github.com/ShobakaBarobaka/ITMO/blob/main/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0%D0%9B%D0%B0%D0%B1%D1%8B/lab4/jsonToXMLlib.py>

Результат программы не отличается от результата самописной программы, однако код теперь представляет пару строк вместо сотни, что не удивительно

1. Дополнительное задание №2

<https://github.com/ShobakaBarobaka/ITMO/blob/main/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0%D0%9B%D0%B0%D0%B1%D1%8B/lab4/jsonToXMLregex.py>

Результат опять же не отличается, так регулярные выражения добавлены с целью упрощения кода программы, а не возвращаемых ей значений

1. Дополнительное задание №3

<https://github.com/ShobakaBarobaka/ITMO/blob/main/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0%D0%9B%D0%B0%D0%B1%D1%8B/lab4/jsonToXMLParcer.py>

Результат остался прежним, но теперь программа способна обрабатывать намного более широкий набор входных данных, в отличии от основного задания, программа которой корректно работает лишь для конкретного расписания

1. Дополнительное задание №4

<https://github.com/ShobakaBarobaka/ITMO/blob/main/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0%D0%9B%D0%B0%D0%B1%D1%8B/lab4/progtime.py>

Результат программы:

With Own Parser: 0.4429943561553955 seconds

With Libraries: 0.4517478942871094 seconds

Without anything: 0.3579568862915039 seconds

With Regular Expressions: 0.4574923515319824 seconds

Как видно, быстрее всего работает базовая программа, что объясняется опять таки ограниченостью сферы ее применения

Далее по скорости следует наш парсер он работает медленее, чем базовое задание, но способен превести в XML почти любой JSON объект

Задание с использованием библиотек выполняется дольше из-за дополнительных методов и проверок в библиотеках, которые я использовал, которые при этом не нужны для задания

Программа с регулярными выражениями выполняется дольше всего потому что библиотека долго проверяет каждую регулярку

# **Заключение**

В ходе выполнения лабораторной работы я узнал о формальных грамматиках, БНФ, о разных форматах файлов и языках разметки. Написал парсер JSON файлов, узнал о библиотеках способных в пару строк кода переводить файлы JSON в XML.

# **Список литературы**

1. [Электронный ресурс]: Википедия. Свободная энциклопедия: <https://ru.wikipedia.org/wiki/%D0%A4%D0%BE%D1%80%D0%BC%D0%B0_%D0%91%D1%8D%D0%BA%D1%83%D1%81%D0%B0_%E2%80%94_%D0%9D%D0%B0%D1%83%D1%80%D0%B0#:~:text=%D0%A4%D0%BE%D1%80%D0%BC%D0%B0%20%D0%91%D1%8D%D0%BA%D1%83%D1%81%D0%B0%20%E2%80%94%20%D0%9D%D0%B0%D1%83%D1%80%D0%B0%20(%D1%81%D0%BE%D0%BA%D1%80.,%D0%BE%D0%BF%D0%B8%D1%81%D0%B0%D0%BD%D0%B8%D1%8F%20%D0%BA%D0%BE%D0%BD%D1%82%D0%B5%D0%BA%D1%81%D1%82%D0%BD%D0%BE%2D%D1%81%D0%B2%D0%BE%D0%B1%D0%BE%D0%B4%D0%BD%D1%8B%D1%85%20%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D1%85%20%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0%D1%82%D0%B8%D0%BA>.
2. <https://habr.com/ru/articles/309242/>