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

Вариант 32 (466468 % 36 = 32)

Выполнил:

Ларионов Владислав Васильевич

Группа P3109

Проверил:

Рыбаков Степан Дмитриевич

Санкт-Петербург 2024

Содержание

[**Задание** 3](#_Toc184146208)

[**Основные этапы вычислений** 5](#_Toc184146209)

[1) Обязательное задание 5](#_Toc184146210)

[2) Дополнительное задание №1 7](#_Toc184146211)

[3) Дополнительное задание №2 8](#_Toc184146212)

[4) Дополнительное задание №3 10](#_Toc184146213)

[5) Дополнительное задание №4 11](#_Toc184146214)

[6) Дополнительное задание №5 12](#_Toc184146215)

[**Вывод:** 13](#_Toc184146216)

[**Источники:** 13](#_Toc184146217)

# **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 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>

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. 1. 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 вычислений**

1. Обязательное задание

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рис 1.1 – код программы основного задания

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рис 1.2 – код программы основного задания

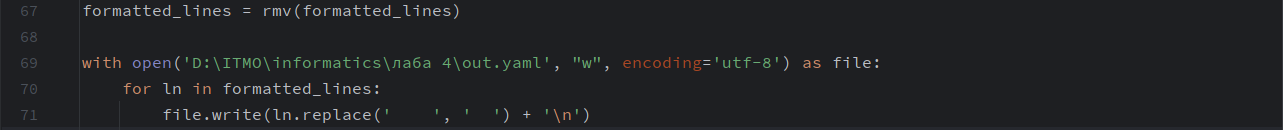


Рис 1.3 – код программы основного задания

1. Дополнительное задание №1

Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, снимок экрана

Автоматически созданное описание

Рис 2 – код программы дополнительного задания

Все данные были перенесены в результирующий формат, однако элементы внутри словарей потеряли порядок (теперь они стоят в алфавитном порядке), но это не коснулось элементов массивов. Порядок элементов словарей не играет значения (в нашем случае), поэтому будем считать парсинг успешным.

1. Дополнительное задание №2

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рис 3.1 – код программы дополнительного задания

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рис 3.2 – код программы дополнительного задания

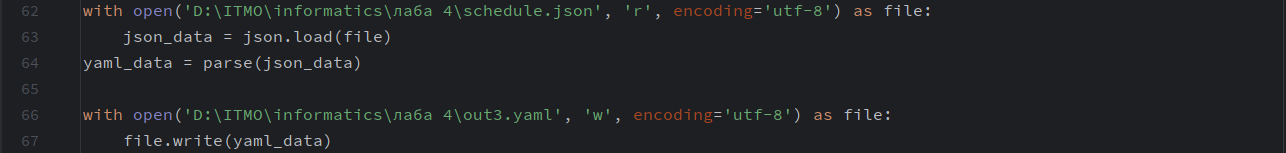


Рис 3.3 – код программы дополнительного задания

Файл был полностью перенесен в результирующий формат, результат парсинга совпадает с результатом парсинга основного задания, поэтому будем считать парсинг успешным.

1. Дополнительное задание №3

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рис 4.1 – код программы дополнительного задания

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рис 4.2 – код программы дополнительного задания

Файл был полностью перенесен в результирующий формат, результат парсинга совпадает с результатом парсинга основного задания, поэтому будем считать парсинг успешным.

1. Дополнительное задание №4

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рис 5 – код программы дополнительного задания

Результат выполнения программы:

Код: 0.15069029999999994

библиотеки: 0.4291378

регулярки: 0.10885309999999992

формальные грамматики: 0.034393600000000024

Парсинг с использованием библиотек показал самый медленный результат, однако это решение обеспечивает гибкость и универсальность. В 3 раза быстрее оказался код, написанный под исходный Json файл, однако этот код может нестабильно работать для других Json файлов. Чуть медленнее этого кода оказался вариант, использующий регулярные выражения, так как происходит работа с шаблонами. Лучший результат показал код, использующий формальные грамматики, благодаря эффективным алгоритмам для сложных структур.

1. Дополнительное задание №5

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Рис 6 – код программы дополнительного задания

CSV – это текстовый формат для хранения табличных данных, где каждая строка представляет запись, а значения разделяются запятыми (или другими разделителями). Он удобен для обмена данными между разными программами, так как поддерживается большинством текстовых редакторов, таблиц и языков программирования. При использовании CSV важно учитывать кодировку, чтобы избежать проблем с символами, особенно в многоязычных данных. CSV не поддерживает вложенные структуры, поэтому не подходит для сложных данных с иерархиями. В Python для работы с CSV-файлами часто применяются библиотеки, такие как csv или pandas, которые упрощают чтение и запись данных.

# **Вывод:**

Во время выполнения данной лабораторной работы я познакомился с форматами обмена данных, различными языками разметки документов. Познакомился с такими форматами, как Json, YAML, CSV. Научился парсить форматы с помощью замены метасимволов, используя регулярные выражения, используя формальные грамматики.

# **Источники:**

1. Балакшин П.В., Соснин В.В. Информатика: методическое пособие. Санкт-Петербург: 2015. (дата обращения: 14.11.2024)
2. Орлов С. А., Цилькер Б. Я. Организация ЭВМ и систем: Учебник для вузов. 2-е изд. – СПб.: Питер, 2011. – 688 с.: ил. (дата обращения: 14.11.2024)
3. Школа системного анализа и проектирования System.Education, <https://systems.education/api-data-formats> (дата обращения: 14.11.2024)