Факультет Программной Инженерии и Компьютерной Техники

Лабораторная работа №4

Исследование протоколов, форматов обмена информацией и

языков разметки документов

Вариант 17

Выполнил:

Козаченко Данил Александрович

Группа P3112

Проверил:

Малышева Т. А.

Доцент ФПИиКТ

Содержание

[Задание 3](#_Toc183044356)

[Основное решение 5](#_Toc183044357)

[Дополнительные задания 5](#_Toc183044358)

[Дополнительное задание №1: 5](#_Toc183044359)

[Дополнительное задание №2: 5](#_Toc183044360)

[Дополнительное задание №3: 5](#_Toc183044361)

[Дополнительное задание №4: 5](#_Toc183044362)

[Дополнительно задание №5: 6](#_Toc183044363)

[Заключение 6](#_Toc183044364)

[Источники 6](#_Toc183044365)

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).
   1. Найти готовые библиотеки, осуществляющие аналогичный парсинг и конвертацию файлов.
   2. Переписать исходный код, применив найденные библиотеки. Регулярные выражения также нельзя использовать.
   3. 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).
    1. Переписать исходный код, добавив в него использование регулярных выражений.
    2. 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).
    1. 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.
    2. Проверку осуществить как минимум для расписания с двумя учебными днями по два занятия в каждом.
    3. 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).
    1. Используя свою исходную программу из обязательного задания и программы из дополнительных заданий, сравнить стократное время выполнения парсинга + конвертации в цикле.
    2. 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).
    1. 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. 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.

Основное решение

Исходный код всех файлов лабораторной работы представлен по ссылке:

<https://github.com/Dkanil/ITMO/tree/main/semester1/informatics/labs/lab4>

Дополнительные задания

Дополнительное задание №1:

Была использована библиотека xmlplain версии 1.6.0. После выполнения программы единственным различием в результатах выполнения программы было то, что при использовании библиотеки xmlplain номер аудитории был перенесён в одинарных кавычках, хотя в исходном XML файле был написан без них. Это различие можно увидеть на Рисунке 1.



Рисунок

Дополнительное задание №2:

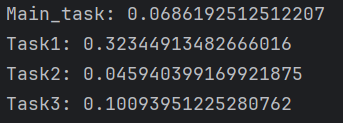
Различий в результатах выполнения программы не было, получен идентичный файл YAML.

Дополнительное задание №3:

Различий в результатах выполнения программы не было, получен идентичный файл YAML.

Дополнительное задание №4:

Стократное время, затраченное на выполнение каждого из заданий представлено на Рисунке 2



Рисунок

Самое быстрое время показало решение с использованием регулярных выражений, так как оно было выполнена без многократных проходов по изначальному файлу. Самыми медленными оказались алгоритмы с использованием формальных грамматик. Это связано с тем, что в таком случае требуется полный перевод первоначального файла в словарь Python, что значительно замедляет программу. Моя реализация парсинга оказалась быстрее реализации из встроенной библиотеки, потому что в моей реализации крайне ограничена функциональность по сравнению с встроенными библиотеками.

Дополнительно задание №5:

Язык TOML является мощным языком, обеспечивающим лёгкую читаемость как для человека, так и для компьютера. Одна из особенностей языка заключается в том, что в нём можно передавать ключам массивы данных. Также TOML позволяет создавать вложенные структуры. Часто этот язык используется для файлов конфигураций.

Заключение

Я изучил такие языки разметки, как XML и YAML. Узнал, что такое формальные грамматики, научился осуществлять парсинг и конвертацию данных из формата XML в YAML. Сравнил скорость работы разных способов конвертации и определил самый быстрый из них.

Источники

1. Балакшин П.В. Информатика Лекция 4, 2024 - <https://t.me/balakshin_students/267>
2. Библиотека xmlplain - <https://pypi.org/project/xmlplain/>
3. Что такое XML - <https://habr.com/ru/articles/524288/>
4. YAML за 5 минут: синтаксис и основные возможности - <https://tproger.ru/translations/yaml-za-5-minut-sintaksis-i-osnovnye-vozmozhnosti>