Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра автоматизированных систем управления (АСУ)

СИНТАКСИЧЕСКИЙ АНАЛИЗ С ИСПОЛЬЗОВАНИЕМ РЕГУЛЯРНЫХ ВЫРАЖЕНИЙ

Отчет по лабораторной работе №2

По дисциплине

«Теория языков программирования и методы трансляции»

Студент гр. 431-3

\_\_\_\_\_\_\_\_\_\_\_\_\_ Е.П. Бекиш

(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата)

Руководитель:

к.т.н., зав. каф. АСУ ТУСУР

\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Романенко

(подпись)

Томск 2024

**Оглавление**

[Введение 3](#_Toc183877464)

[1 ТЕОРИЯ 4](#_Toc183877465)

[1.1 Синтаксис описания переменных 4](#_Toc183877466)

[1.2 Построение регулярного выражения 4](#_Toc183877467)

[2 РЕЗУЛЬТАТЫ РАБОТЫ 6](#_Toc183877468)

[2.1 Программная реализация 6](#_Toc183877469)

[2.2 Тестирование программы 6](#_Toc183877470)

[Заключение 8](#_Toc183877471)

[Список использованных источников 9](#_Toc183877472)

[Приложение А (обязательное) Регулярное выражение 10](#_Toc183877473)

[Приложение Б (обязательное) Листинг программы 11](#_Toc183877474)

# Введение

Цель: научиться применять на практике такие средства синтаксического анализа, регулярных выражений.

Задание: написать программу, которая должна читать входные данные из текстового файла (например, имеющего имя «input.txt»), и выдавать результат работы в текстовый файл (например, имеющий имя «output.txt»). Для ввода и вывода данных допускается использование в программе визуального интерфейса вместо файлового ввода/вывода.

Вариант 1. На вход программы подается описание переменных на выбранном языке (Pascal, C++, C# и т.д.). Программа должна проанализировать его при помощи ДКА или ДМПА и выдать результат проверки. Это может быть:

1. Сообщение о том, что описание корректное.
2. Сообщение о синтаксической ошибке. Указывать тип ошибки не обязательно, требуется указать строку и позицию в строке входного файла, где находится ошибка.
3. Сообщение о дублирование имен переменных. В этом случае на выходе программы необходимо указать имя дублируемой переменной, строку и позицию в строке, где находится дубликат.

# ТЕОРИЯ

В данном разделе приведена краткая теория, которая использовалась в ходе выполнения программной части лабораторной работы.

## 1.1 Синтаксис описания переменных

В качестве языка программирования был выбран С++, описание переменных которого подавалось на вход программы. Правила описания переменных в таком случае включают следующие аспекты:

* имя переменной может состоять только из латинских букв, цифр и символа подчеркивания;
* имя переменной не может начинаться с цифры;
* имя переменной не может повторяться, то есть нельзя объявить две переменные с одним именем;
* в качестве имени переменной не могут использоваться ключевые слова языка C++. Поддерживаемый список таких слов, следующий: int, double, long, short, char, float.

Таким образом, описание состоит из указания типа данных со следующими за ним списком имён переменных. Помимо этого, поддерживаются модификаторы размера типа: long, short. При этом модификатор может использоваться без указания базового типа. В этом случае в качестве базового типа подразумевается тип int. Модификатор long может использоваться с типами int и double, модификатор short — только с типом int. Без данных модификаторов используются типы float, char. Также, в программе поддерживается описание многомерных статических массивов, в качестве размеров которого могут указывается только натуральные целые числа.

## Построение регулярного выражения

В качестве способа определения языка был выбран метод непосредственной записи регулярного выражения, вместо его получения из решения системы уравнений с регулярными коэффициентами. Выбор был сделан по той причине, что количество состояний, полученное в ходе составления функции переходов ДКА составило 17 состояний, что затрудняет решение системы уравнений вручную.

Первым делом была определена начальная конструкция, которая включает в себя именованные группы с правильными цепочками:

* (?<type>int(?:\s+long|\s+short)?|double(?:\s+long)?|long(?:\s+int|\s+double)?|short(?:\s+int)?|float|char) — именованная группа type, которая захватывает все правильные комбинации типа данных и его модификатора;
* (?<id>[a-zA-Z\_][a-zA-Z\_0-9]\*) — именованная группа id, захватывающая все правильные цепочки имени идентификатора;
* (?<op>[1-9]\d\*) - именованная группа op, которая захватывает правильно указанный размер массива.

# 2 РЕЗУЛЬТАТЫ РАБОТЫ

## 2.1 Программная реализация

В качестве языка программирования был выбран python версии 3.12.7. Вспомогательной библиотекой, реализующей управляющее устройство, явилась библиотека regex. Графический интерфейс, поддерживающий считывание входного сообщения как вручную, так и из файла, был обеспечен библиотекой streamlit.

После считывания данных для анализа, они отправляются в функцию, в которой происходит итерация по соответствиям шаблону, возвращённым методом finditer библиотеки regex. Уже внутри данного цикла запускается цикл, итерирующий по группе id.

Реализация лабораторной работы представлена в листинге В.1.

## 2.2 Тестирование программы

В ходе тестирования программы было проверено три ситуации, результаты которых представлены на рисунках 2.1 – 2.3, где вызываются исключения, а так же и корректные ситуации.

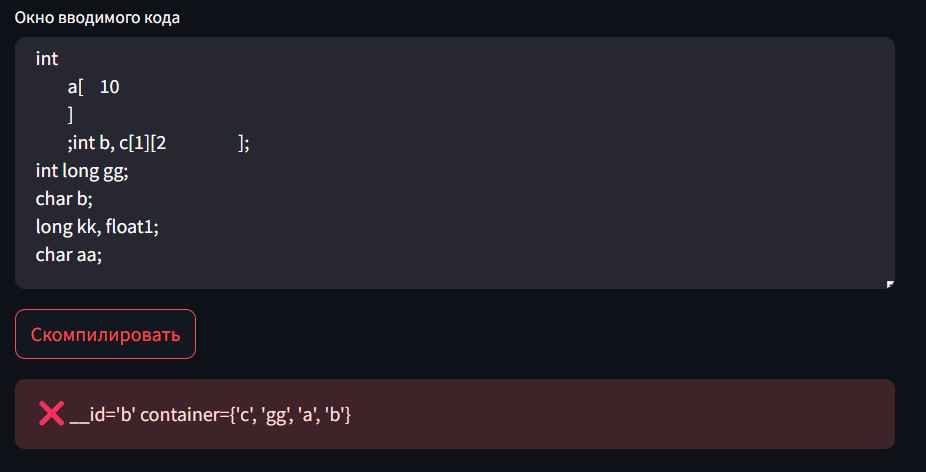


Рисунок 2.1—Тестирование файла с повторным id

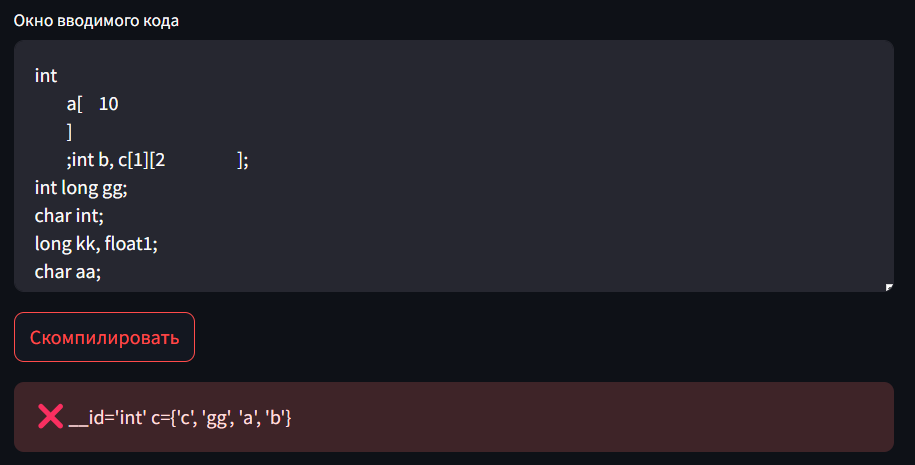


Рисунок 2.2 — Тестирование файла с id, который является типом

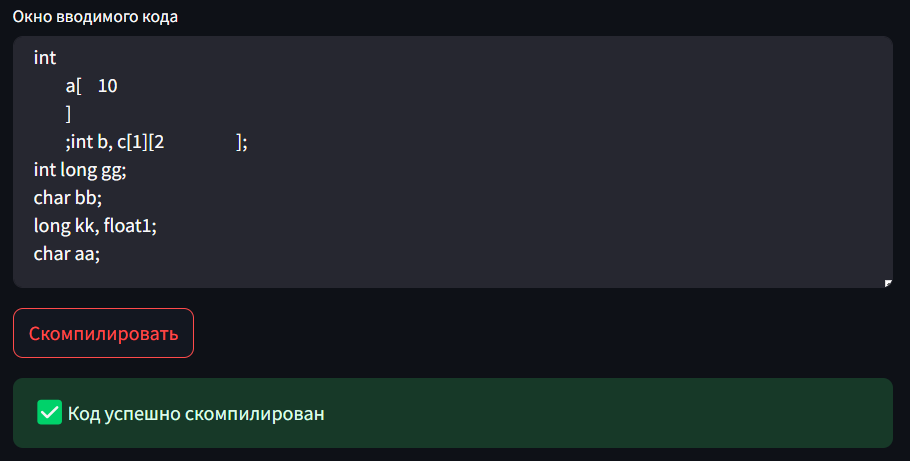


Рисунок 2.3 — Успешное завершение анализа описания переменных

# Заключение

В результате выполнения лабораторной работы я научился применять на практике такие средства синтаксического анализа, как регулярные выражения.

# Список использованных источников

1. Романенко, В. В. Теория языков программирования и методы трансляции: Учебное пособие [Электронный ресурс] / В. В. Романенко, В. Т. Калайда. — Томск: ТУСУР, 2019. — 264 с. — Режим доступа: https://edu.tusur.ru/publications/9043
2. Романенко, В. В. Теория языков программирования и методы трансляции: Учебно-методическое пособие по выполнению лабораторных работ [Электронный ресурс] / В. В. Романенко, В. Т. Калайда. — Томск: ТУСУР, 2019. — 122 с. — Режим доступа: https://edu.tusur.ru/publications/9044
3. Образовательный стандарт вуза ОС ТУСУР 01-2021. Работы студенческие по направлениям подготовки и специальностям технического профиля. Общие требования и правила оформления от 25.11.2021 [Электронный ресурс]. – Режим доступа: <https://regulations.tusur.ru/documents/70>.

# Приложение А (обязательное) Регулярное выражение

Листинг А.1 — Исходный текст регулярного выражения

\s\*

(

int(?:\s+long|\s+short)?|

double(?:\s+long)?|

long(?:\s+int|\s+double)?|

short(?:\s+int)?|

float|

char

)

\s+

(

(?:

(?<id>

[\_a-zA-Z][\_a-zA-Z0-9]\*

)

(?:

\s\*\[

\s\*

[1-9]\d\*

\]

\s\*

)\*

)

\s\*

(

,

(?=

\s\*

[\_a-zA-Z]

)|

;

)

\s\*

)\*

# Приложение Б (обязательное) Листинг программы

Листинг Б.1 — Содержимое файла main.py

import regex

file\_regex = regex.compile(open('reg.txt').read())

container = set()

for a in regex.finditer(file\_regex, open('syntax.txt').read(), partial=True):

\_\_ids = a.capturesdict()['id']

if \_\_ids:

for \_\_id in \_\_ids:

if \_\_id in container or \_\_id in ['int', 'float', 'double', 'char', 'long', 'short']:

print(f'{\_\_id=}')

raise Exception()

container.add(\_\_id)

print(f'{container=}')