

# Лабораторная работа № 1.5 «Порождение лексического анализатора с помощью flex»

Скоробогатов С.Ю., Коновалов А.В.

5 апреля 2016

## 1 Цель работы

Целью данной работы является изучение генератора лексических анализаторов flex.

## 2 Задание

Требуется воспользоваться генератором лексических анализаторов flex для порождения лексического анализатора на языке C (или C++).

Порождённый лексический анализатор должен загружать входной поток из файла (кодировка — ASCII) и выводить в стандартный поток вывода описания распознанных лексем в формате

Тег (координаты\_фрагмента): атрибут лексемы

При этом для лексем, не имеющих атрибутов, нужно выводить только тег и координаты. Например,

```
IDENT (1, 2)-(1, 4): str
ASSIGN (1, 8)-(1, 9):
STRING (1, 11)-(1, 16): qwerty
```

Лексемы во входном файле могут разделяться пробельными символами (пробел, горизонтальная табуляция, маркеры окончания строки), а могут быть записаны слитно (если это не приводит к противоречиям).

Идентификаторы и числовые литералы не могут содержать внутри себя пробельных символов, если в задании явно не указано иного. Комментарии, строковые и символьные литералы могут содержать внутри себя пробельные символы.

Входной файл может содержать ошибки, при обнаружении которых лексический анализатор должен выдавать сообщение с указанием координаты, восстанавливаться и продолжать работу.

Для лексических доменов должны вычисляться их атрибуты (если указанный лексический домен есть в варианте задания):

- для целых чисел атрибут должен быть целым числом наибольшей разрядности (например, в Java — long).
- для вещественных чисел атрибут должен быть вещественным числом (например, double в Java или C++),
- для идентификаторов — номер в таблице идентификаторов (см. слайды лекции),
- для строковых констант — значение, изображаемое самой строковой константой (т.е. без окружающих кавычек и с интерпретацией escape-последовательностей, если таковые есть в варианте задания),
- для комментариев токен не порождается, вместо этого координаты комментария помещается в список комментариев (см. слайды лекции).

## 3 Индивидуальный вариант

Строковые литералы: ограничены двойными кавычками, могут содержать Escape-последовательности «\"», «\n», «\t» и «\\», не пересекают границы строк текста. Числовые литералы: последовательности десятичных цифр, разбитые точками на группы по три цифры («100», «1.000», «1.000.000»). Идентификаторы: последовательности латинских букв, знаков подчёркивания и цифр, начинающиеся с буквы или подчёркивания.

## 4 Отчёт по лабораторной работе № 1.5

Отчёт выполняется в разметке Markdown по следующему шаблону:

```
% Лабораторная работа № 1.5 «Порождение лексического анализатора с помощью flex»
% 4 апреля 2023 г.
% Вася Пупкин, ИУ9-63Б
# Цель работы
<переписываете цель работы из задания>
# Индивидуальный вариант
<переписываете индивидуальный вариант>
# Реализация
```Lex
...
```
# Тестирование
Входные данные
```
...
```
Вывод на `stdout`
```
...
```
# Вывод
<пишете, чему научились>
```

В отчёте приведён лишь необходимый минимум.

[Шаблон отчёта](#)

Ваш отчёт будет конвертирован в PDF при помощи pandoc следующей командой:

```
pandoc \
--pdf-engine=xelatex \
-V 'mainfont:Liberation Serif' \
-V 'monofont:Liberation Mono' \
"$SOURCE" -o "$PDF"
```

Язык реализации: Markdown

Код решения

Из файла

Отправить