Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный технический университет имени Н. Э. Баумана (национальный исследовательский университет)

Факультет «Робототехника и комплексная автоматизация»

Кафедра «Системы автоматизированного проектирования»

**Отчет по лабораторной работе № 1**

По курсу «Программное обеспечение систем автоматизированного проектирования»

Выполнил: Студент Гусаров А. А.

Группа РК6-33Б

Проверил:

Дата \_\_\_\_\_\_\_\_\_\_\_\_

Подпись \_\_\_\_\_\_\_\_\_\_\_\_

Москва, 2020 г.

## Введение

Настоящий документ определяет техническое задание на разработку программы лексического анализатора адресов (далее – ЛАА) для поиска во входном потоке записей адресов IP (Internet Protocol) класса С.

## Основания для разработки

Программа ЛАА разрабатывается в рамках лабораторной работы по курсу "*Лингвистическое обеспечение САПР* " для практического изучения этапа лексического анализа в процедурах трансляции формальных языков.

## Назначение разработки

Программа ЛАА предназначается для обработки адресов IP с целью автоматизации нахождения корректных IP адресов класса С.

## Требования к программе

1. ***Требования к функциональным характеристикам***
   1. Программа ЛАА должна осуществлять поиск IP адресов по указанным правилам из входного потока стандартного ввода символьных данных и отображать их в потоке стандартного вывода.
   2. Каждый поиск IP адресов, выделенный программой ЛАА, должен завершаться символом перевода строки ('\n') в потоке стандартного вывода (вывод в одну колонку).

## Требования к надежности

1. ***Требования к составу и параметрам технических средств***

Программа ЛАА должна быть разработана исходя из возможности реализации на стандартном составе технических средств компьютеров любой архитектуры, после соответствующей трансляции исходного кода.

## Условия эксплуатации

* 1. Программа ЛАА должна быть ориентирована на эксплуатацию в среде OS UNIX
  2. Программа ЛАА должна быть реализована в виде выполняемого файла с именем a.out, по которому она должна вызываться средствами любого командного процессора OS UNIX.
  3. Программа ЛАА должна эксплуатироваться в режиме фильтра, который читает поток стандартного ввода и отображает результаты в потоке стандартного вывода OS UNIX.

## Требования к информационной и программной совместимости

* 1. При разработке программы ЛАА необходимо использовать генератор лексических анализаторов (далее по тексту - LEX) из состава инструментальных средств OS UNIX, который ориентирован на обработку

файла спецификаций (далее по тексту - Lex-файл) проектируемого лексического анализатора.

* 1. При разработке программы ЛАА необходимо составить Lex-файл, отражающий специфику лексического анализа чисел, и сохранить его под именем Lab1.l в выбранном рабочем каталоге файловой системы OS UNIX.
  2. Lex-файл Lab1.l программы ЛАА может содержать 3 раздела:
* раздел деклараций (необязательный раздел);
* раздел правил (обязательный раздел);
* раздел функций (необязательный раздел),

которые должны разделять символьные пары %%.

* 1. В разделе деклараций Lex-файла программы ЛАА необходимо:
* объявить макрошаблоны ограничителей значений старшего байта и младших байтов SENIOR и JUNIOR, которые будут использоваться в разделе правил LEX-файла для управления активность правил;
  1. В разделе правил Lex-файла программы ЛАА должны быть введены правила, которые обеспечивают распознавание необходимых лексем с помощью шаблонов регулярных выражений и их функциональную обработку посредством блоков действий.
  2. Правила раздела правил Lex-файла программы ЛАА должны включать шаблоны регулярных выражений, которые обеспечивают распознавание следующих лексем:
* старший и младшие байты (по макрошаблонам {SENIOR} и {JUNIOR});
* символ '.', разделяющий десятичные коды;
* любые символы вне IP адреса.
  1. Правила раздела правил Lex-файла программы ЛАА должны обеспечивать выполнение следующих действий при распознавании лексем, перечисленных в п. 5.6:
* отображение в поток стандартного вывода подходящего значения путем печати содержания символьного массива yytext библиотечной функцией printf системы программирования C;
* игнорировать стандартный вывод лексем, не содержащих в себе IP адресов через реализацию пустого действия;
  1. В разделе функций Lex-файла Lab1.l программы ЛАА в основной функции main() необходимо обеспечить вызов функции yylex(), формируемой LEX по Lex-файлу для обработки лексем, заявленной в разделе правил.

## Стадии и этапы разработки

В процессе разработки программы ЛАА средствами LEX необходимо выполнить следующие этапы:

* подготовить файл спецификаций лексического анализатора (Lex- файл) для программы ЛАА с именем Lab1.l, используя любой текстовый редактор OS UNIX, например, nano;
* получить исходный С-код программы ЛАА в файле Lab1.c, обработав Lex-файл Lab1.l командой lex следующим образом:

# $ lex -t Lab1.l > Lab1.c

* построить по исходному коду Lab1.c выполняемый файл Lab1 программы ЛАА, вызвав C-компилятор OS UNIX и подключив стандартную библиотеку объектных модулей LEX libl.a следующим образом:

# $ cc -o Lab1 Lab1.c -ll

## Порядок контроля и приемки

1. Для контроля функционирования программы ЛАА необходимо разработать контрольный пример, содержащий различные варианты численных значений, сохранив его в текстовом файле Ex.txt..
2. Для реализации контроля программы ЛАА по контрольному примеру необходимо использовать следующий набор команд OS UNIX:

# $ ./Lab1 < Ex.txt

который обеспечивает перенаправление ввода из текстового файла в ЛАА.

## Приложение

При разработке ЛА в OS UNIX использовались следующие источники, указанные ниже.

1. Волосатова Т. М. «Лингвистическое обеспечение САПР» // Режим доступа : http://bigor.bmstu.ru/?cnt/?doc=LO-SAPR/base.cou (дата обращения 26.09.2020).
2. Компаниец Р. И., Маньков Е. В., Филатов Н. Е.. Системное программирование. Основы построения трансляторов, СПб.: КОРОНА принт, 2004.
3. Тихомиров В. П., Давидов М. И.. Компиляторы: принципы, технологии и инструментарий, 2-е изд.: Пер. С англ. - М.: ООО «И.Д. Вильямс», 2008.