

Пример отчета по лабораторным работам по автоматизированной разработке лексических анализаторов

Лексический анализатор комментариев

Задание

Текст задания: разработать лексический анализатор комментариев в формате системы программирования С.

Введение

Настоящий документ определяет техническое задание на разработку программы лексического анализатора комментариев (далее по тексту - ЛАК) в формате системы программирования С.

Основания для разработки

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

Назначение разработки

Программа ЛАК предназначена для обработки комментариев исходных текстов программ, составленных в системе программирования С, с целью автоматизации процедуры подготовки программной документации.

Требования к программе

1. Требования к функциональным характеристикам

1.1. Программа ЛАК должна выделять блоки комментариев системы программирования С из входного потока стандартного ввода символьных данных и отображать их в потоке стандартного вывода.

1.2. Блоки комментариев, распознаваемые программой ЛАК, должны быть ограничены символами `/*` и `*/`.

1.3. Программа ЛАК должна обеспечивать выделение как одно-строчных:

`/* Это однострочный комментарий */`

так и многострочных:

`/* Это много- строчный комментарий */`

блоков комментариев, допустимых в системе программирования С.

1.4. Программа ЛАК должна подавлять в выходном потоке стандартного вывода любые символы входного потока, расположенные вне блоков комментариев, и ограничители блоков комментариев, указанные в п. 1.2.

1.5. Каждый блок комментариев, выделенный программой ЛАК, должен завершаться символом перевода строки ('\n') в потоке стандартного вывода.

2. Требования к надежности

Программа ЛАК должна игнорировать получение сигналов прерывания и выхода от клавиатуры.

3. Требования к составу и параметрам технических средств

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

4. Условия эксплуатации

4.1. Программа ЛАК должна быть ориентирована на эксплуатацию в среде OS UNIX

4.2. Программа ЛАК должна быть реализована в виде выполняемого файла с именем `lsc`, по которому она должна вызываться средствами любого командного процессора OS UNIX.

4.3. Программа ЛАК должна эксплуатироваться в режиме фильтра, который читает поток стандартного ввода и отображает результаты в потоке стандартного вывода OS UNIX.

5. Требования к информационной и программной совместимости

5.1. При разработке программы ЛАК 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.

5.2. При разработке программы ЛАК необходимо составить Lex-файл, отражающий специфику лексического анализа комментариев, и сохранить его под именем `lsc.l` в выбранном рабочем каталоге файловой системы OS UNIX.

5.3. Lex-файл `lsc.l` программы ЛАК должен содержать 3 раздела:

- раздел деклараций;
- раздел правил;
- раздел функций,

которые должны разделять символьные пары `%%`.

5.4. В разделе деклараций Lex-файла программы ЛАК необходимо:

- объявить метку предусловия обработки блока комментария `ISCOM`, используя директиву `%Start` в первой строке Lex-файла;
- объявить макрошаблоны ограничителей комментариев `COMOPEN` и `CLOSECOM`, которые будут использоваться в разделе правил Lex-файла для управления активностью правил;

- описать блок внешних переменных (extern), макроопределений (define) и заголовочных файлов (include), которые необходимы в действиях раздела правил, ограничив его директивами %{ и %}.

5.5. В разделе правил Lex-файла программы ЛАК должны быть введены правила, которые обеспечивают распознавание необходимых лексем с помощью шаблонов регулярных выражений и их функциональную обработку посредством блоков действий.

5.6. Правила раздела правил Lex-файла программы ЛАК должны включать шаблоны регулярных выражений, которые обеспечивают распознавание следующих лексем:

- начало и конец комментария (по макрошаблонам {COMOPEN} и {CLOSECOM});
- любые символы, кроме символа "*", внутри блока комментария (по предусловию);
- любой символ, кроме символа "/", после символа "*" внутри блока комментария (по предусловию);
- любые символы вне блока комментария.

5.7. Правила раздела правил Lex-файла программы ЛАК должны обеспечивать выполнение следующих действий при распознавании лексем, перечисленных в п. 5.6:

- установка и инициализация метки предусловий ISCOM с помощью оператора BEGIN при обнаружении начала и конца комментария, соответственно;
- отображение в поток стандартного вывода выделенной внутри комментария лексемы с помощью макроса ECHO или путем печати содержания символьного массива yytext библиотечной функцией printf системы программирования C;
- игнорировать стандартный вывод лексем вне блока комментария через реализацию пустого действия;

5.8. В разделе функций Lex-файла lsc.l программы ЛАК необходимо специфицировать основную функцию main(), которая должна установить игнорирование клавиатурных сигналов, указанных в подразделе 2, и обеспечить вызов функции yylex(), формируемой LEX по Lex-файлу для обработки лексем, заявленной в разделе правил. Специфицированная функция main() будет перегружать стандартный вариант основной функции лексического анализатора из библиотеки объектных модулей LEX

5.9. В Lex-файле программы ЛАК должно быть указано соответствующее число комментариев в формате системы программирования C, которое необходимо для составления его описания в виде пояснительной записки

Требования к программной документации

1. По результатам разработки необходимо составить пояснительную записку в текстовом файле lac.lex, содержащую описание Lex-файла.

2. Для автоматизации составления пояснительной записки необходимо обработать комментарии Lex-файла lex.l программой ЛАК, используя следующий конвейер команд OS UNIX:

```
$ cat lex.l | lac > lac.lex
```

и любой текстовый редактор (например, xedit) для минимальной пост-корректировки содержания полученного описания в файле lac.lex.

Стадии и этапы разработки

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

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

```
$ lex -t lac.l > lac.c
```

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

```
$ cc -o lac lac.c -ll
```

Порядок контроля и приемки

1. Для контроля функционирования программы ЛАК необходимо разработать контрольный пример, содержащий различные варианты компоновки блоков комментариев, сохранив его в текстовом файле lac.chk.

2. Для реализации контроля программы ЛАК по контрольному примеру необходимо использовать следующий конвейер команд OS UNIX:

```
$ cat lac.chk | lac | more
```

который обеспечивает постраничную или построчную визуализацию результатов обработки комментариев в файле lac.chk.

Приложение

При разработке лексических анализаторов в OS UNIX рекомендуется использовать литературные источники, перечисленные ниже.

1. Кристиан К. Введение в операционную систему UNIX, М.: Финансы и статистика, 1985.
2. Тихомиров В.П., Давидов М.И. Операционная система ДЕМОС: инструментальные средства программирования, М.: Финансы и статистика, 1988.
3. SCO XENIX, Development System, LEX Programmer Guide, SCO Inc., 1986. (Имеется русский перевод: Генератор лексических анализаторов LEX. Руководство Программиста)