LEX

Lex is a computer program that generates lexical analysers. Lex is commonly used with the yacc parser generator. Lex, originally written by Mike Lesk and Eric Schimdt and described in 1975, is the standard lexical analyzer generator on many Unix systems, and an equivalent tool is specified as part of the POSIX standard. Flex (fast lexical analyzer generator) is a free and open-source software alternative to lex.

Lex reads an input stream specifying the lexical analyzer and writes source code which implements the lexical analyzer in the C programming language.

A lex program consists of three sections: a section containing *definitions*, a section containing *translations*, and a section containing *functions*.

Throughout a lex program, you can freely use newlines and C-style comments; they are treated as white space. Lines starting with a blank or tab are copied through to the lex output file. Blanks and tabs are usually ignored, except when you use them to separate names from definitions, or expressions from actions.

The definition section is separated from the following section by a line consisting only of **%%**. In this section, named regular expressions can be defined, which means you can use names of regular expressions in the translation section, in place of common subexpressions, to make that section more readable. The definition section can be empty, but the **%%** separator is required.

The translation section follows the definition section, and contains regular expressions paired with actions, which describe what the lexical analyzer is to do when a match of a given regular expression is found. The first nonescaped space or tab on a line in the translation section signals the start of the action. Actions are further described in later sections of this topic.

You can omit the function section; if it is present, it is separated from the translation section by a line containing only **%%**. This section can contain anything, because it is simply attached to the end of the lex output file.

Installation

To install lex in Ubuntu, run the following command:

```
sudo apt install flex
```

Example

```
/*** Definition section ***/

%{
/* C code to be copied verbatim */
#include <stdio.h>
%}

%%
```

```
    /*** Rules section ***/

    /* [0-9]+ matches a string of one or more digits */
[0-9]+  {
            /* yytext is a string containing the matched text. */
            printf("Saw an integer: %s\n", yytext);
        }

.|\n    {   /* Ignore all other characters. */   }

%%
/*** C Code section ***/

int main(void)
{
    /* Call the lexer, then quit. */
    yylex();
    return 0;
}
```

If this input is given to `flex`, it will be converted into a C file, `lex.yy.c`. This can be compiled into an executable which matches and outputs strings of integers. For example, given the input:

```
abc123z.!&*2gj6
```

the program will print:

```
Saw an integer: 123
Saw an integer: 2
Saw an integer: 6
```