**Practical 2**

**Aim:** Count the number of comments, keywords, identifiers, words, lines and spaces from input file.

**Theory:** The aim of this program is to analyse a source code input using LEX and count the number of comments, keywords, identifiers, words, lines, spaces, tabs, characters.

LEX is a tool used to generate scanners, also known as tokenizers, which are programs that recognize lexical patterns in text. In this program, LEX is used to scan through the source code and identify different lexical components based on regular expressions.

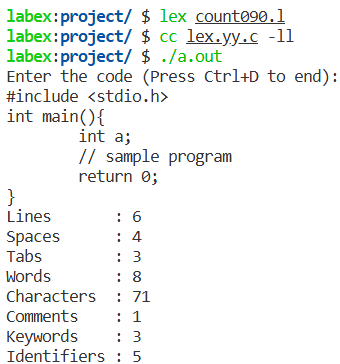
Key Components:

* Comments: Both multi-line (/\* ... \*/) and single-line (// ...) comments are detected and counted.
* Keywords: A static array of C language keywords is maintained. Any token matching these is identified as a keyword.
* Identifiers: Tokens that are valid C identifiers and not keywords are counted as identifiers.
* Words: Every valid identifier or keyword is treated as a word.
* Lines: Every newline character (\n) increments the line counter.
* Spaces and Tabs: Detected using " " and "\t" respectively.
* Characters: Total length of all recognized tokens is added to this count.

**Code:**



**Output:**



**Conclusion:** This LEX program successfully reads a source code input and provides a lexical analysis report by accurately counting, total lines, spaces, tabs, and characters, number of C language keywords and identifiers, total number of comments, word count based on identifiers and keywords.