Exp No: 3

Date:

DEVELOP A LEXICAL ANALYZER TO RECOGNIZE TOKENS USING LEX TOOL

AIM:

To implement the program to identify C keywords, identifiers, operators, end statements like [], {} using LEX tool.

ALGORITHM:

- 1. Initialize a variable n to count the number of lines.
- 2. Define patterns for letters, digits, identifiers, arithmetic operators (AO), relational operators (RO), preprocessor directives (pp), and other symbols.
- 3. Define actions to perform when a pattern is matched and display the corresponding pattern type.
- 4. Open the file "sample.c" for reading and invoke lexical analysis with yylex().
- 5. Count the number of newline characters encountered and store it in n.
- 6. Display the number of lines, n.

PROGRAM:

```
%option noyywrap
letter [a-zA-Z]
digit [0-9] id
[a-zA-Z]
AO [+|-|/|%|*]
RO [<|>|<=|>=| pp
[#]
%{
int n=0;
%}
%%
"void"
                                printf("%s return type\n",yytext);
                                printf("%s Function\n",yytext);
{letter}*[(][)]
"int"|"float"|"if"|"else" printf("% keywords\n",yytext);
"printf"
                                        printf("%s keywords\n",yytext);
{id}((id)|(digit))*
                                printf("%s Identifier\n",yytext);
```

Roll Number: 210701107

Name: S.Karthic

```
{digit} {digit}*
                                 printf("%d Numbers\n",yytext);
{AO}
                                         printf("%s Arithmetic Operators\n",yytext);
                                         printf("%s Relational Operators\n",yytext);
{RO}
{pp}{letter}*[<]{letter}*[.]{letter}[>] printf("%s processor
Directive\n", yytext);
\lceil n \rceil
                                         n++;
"."|","|"}"|"{"|";"
                                 printf("%s others\n",yytext);
%%
int main(){
        yyin=fopen("sample.c","r");
yylex();
        printf("No of Lines %d\n",n);} OUTPUT:
```

```
-(kali®kali)-[~/Documents/cdlab]
─$ vi exp2.l
(kali@kali)-[~/Documents/cdlab]
$ lex exp2.l
  -(kali®kali)-[~/Documents/cdlab]
gcc lex.yy.c
   -(kali®kali)-[~/Documents/cdlab]
int a = b + c;
int keywords
 a Identifier
 = Relational Operators
b Identifier
 + Arithmetic Operators
 c Identifier
; others
float t = 0.5 * a;
float keywords
t Identifier

    Relational Operators

1741780218 Numbers
. others
1741780220 Numbers
 * Arithmetic Operators
 a Identifier
; others
```

RESULT:

Thus, a c program is implemented to identify C keywords, identifiers, operators, end statements like [], {} using LEX tool.

Roll Number: 210701107

Name: S.Karthic