

Register Number – 21BPS1045

Name – SAJAL BRAHMA

AIM : Design a LEX program to identify numbers , keyword (more than 10)
identifiers , header files , parenthesis.

PROGRAM CODE -

```
%{
#include<stdio.h>
#include<stdlib.h>
}%

DIGIT [0-9]
LETTER [a-zA-Z]
IDENTIFIER {LETTER}({LETTER}|{DIGIT})*
NUMBER {DIGIT}+(\.{DIGIT}+)?(E[+-]?{DIGIT}+)?
WHITESPACE [ \t\n]

%%
{WHITESPACE} ; // Ignore whitespace
{IDENTIFIER} { printf("Identifier: %s\n", yytext); }
{NUMBER} { printf("Number: %s\n", yytext); }
"+" { printf("Operator: %s\n", yytext); }
"-" { printf("Operator: %s\n", yytext); }
"*" { printf("Operator: %s\n", yytext); }
"/" { printf("Operator: %s\n", yytext); }
"=" { printf("Operator: %s\n", yytext); }
"(" { printf("Left Parenthesis: %s\n", yytext); }
")" { printf("Right Parenthesis: %s\n", yytext); }
";" { printf("Semicolon: %s\n", yytext); }
. { printf("Unknown token: %s\n", yytext); }
%%

int main()
{
    yylex();
    return 0;
}

int yywrap()
{
    return 1;
}
```

OUTPUT -

```
student@admin: ~/Desktop
File Edit View Search Terminal Help
student@admin:~$ cd Desktop
student@admin:~/Desktop$ flex lex.l
student@admin:~/Desktop$ gcc -o test-ll lex.yy.c
student@admin:~/Desktop$ ./test-ll
int a = 10
Identifier: int
Identifier: a
Operator: =
Number: 10
float int abc = 25
Identifier: float
Identifier: int
Identifier: abc
Operator: =
Number: 25
█
```

INFERENCE -

The code is successfully implemented and verified in the lab.