## Date:- 20/7/2023

## Roll No and Name :- 20BCE075 , Yash Ginoya

## Course code and Name :- 2CS701, COMPILER CONSTURCTION

## Practical No. :- 1

**AIM :-** TO IMPLEMENT LEXICAL ANALYSER TO RECOGNIZE ALL DISTINCT TOKEN CLASSES: USE FLEX/LEX TOOL TO RECOGNIZE ALL DISTINCT TOKEN CLASSES (DATA TYPE, IDENTIFIER, CONSTANT (INTEGER, FLOAT,CHAR, STRING), OPERATOR (ARITHMETIC, RELATIONAL, ASSIGN, UNARY
+/-, INCREMENT), SINGLE LINE/MULTI-LINE COMMENTS, SPECIAL SYMbOL(;,{}())) .

GENERATE LEXICAL ERROR REPORTS FOR INVALID LEXEME.

**CODE :-**

```
% {
#include <stdio.h>
      % } %
    % "\n" return 0;
[0 - 9] + { printf("NUM"); }
[0 - 9] * "."[0 - 9] + { printf("FNUM"); }
[_a - zA - Z][_a - zA - Z0 - 9] *
{ printf("Identifier"); }
        [- += * / ]
{
    printf("%c\n", yytext[0]);
}
\/\/ [^\n] *
{ /* Ignore single-line comment */ }
\/\*([^*] |\* [^\/ ]) *\*\/ {/* Ignore multi-line comment */}[;, {}()]
{
    printf("Spacial symbol");
}
. { printf("Lexical Error: Invalid Lexeme: %s\n", yytext); }
% %
    int main()
{
```

```
    yylex();
    return 0;
}
int yywrap()
{
    return 0;
}
int yyerror()
{
    return 0;
}
```

## OUTPUT :-

```
PS D:\collage\sem_7\CC\lab\practical 1> flex 20BCE085_CC_Practical_1.l
PS D:\collage\sem_7\CC\lab\practical 1> gcc lex.yy.c
PS D:\collage\sem_7\CC\lab\practical 1> ./a
a=10
Identifier=
NUM
PS D:\collage\sem_7\CC\lab\practical 1>
```

```
PS D:\collage\sem_7\CC\lab\practical 1> ./a
a=10*10
Identifier=
NUM*
NUM
PS D:\collage\sem_7\CC\lab\practical 1>
```