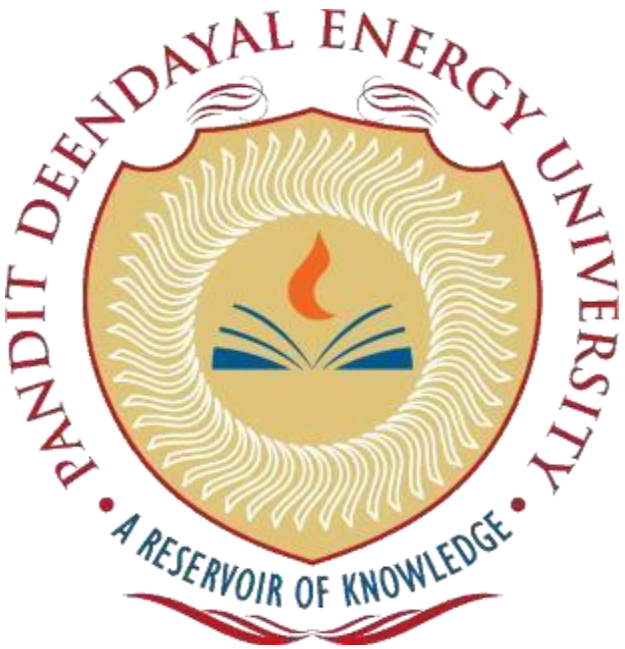
Pandit Deendayal Energy University, Gandhinagar

School of Technology

Department of Computer Science & Engineering

System Software & Compiler Design Lab (20CP302P)



Name: **Sutariya Smit Dharmendrabhai**

Enrolment No: **21BCP142**

Semester: **V**

Division: **3 (G5)**

Branch: **Computer Science Engineering**

Practical: 2

Aim:

a. Write a LEX program to count the number of tokens and display each token with its length in the given statements.

Code:

```
%option noyywrap

%{
   int c1 = 0, c2 = 0, c3 = 0, c4 = 0;
%}

digit [0-9]
letter [a-zA-Z]

%%

auto|break|case|char|const|continue|default|do|double|else|enum|extern|float|for|goto|if|int|long|register|return|short|signed|sizeof|static|struct|switch|typedef|union|unsigned|void|volatile|while {printf("The length of keyword %s: %d \n",yytext, yyleng);c1++;}
```

```
{letter}({letter}|{digit})* {printf("The length of identifier %s is: %d \n",
yytext,yyleng); c2++;}

{digit}+ {printf("The length of digit %s is: %d\n", yytext, yyleng); c3++;}

[ \t\n]

. {printf("The length of others %s is: %d\n", yytext, yyleng); c4++;}
%%

int main() {
   yylex();
   printf("keywords: %d, identifiers: %d, digits: %d, others: %d,\n Total number of
tokens: %d", c1, c2, c3, c4, c1+c2+c3+c4);
   return 0;
}
```

Output:

```
PS D:\Sem-5\compiler> flex Lab_2\prog2a.l
PS D:\Sem-5\compiler> gcc lex.yy.c
PS D:\Sem-5\compiler> ./a

void main() {
The length of keyword void: 4
The length of identifier main is: 4
The length of others ( is: 1
The length of others ) is: 1
The length of others { is: 1
    float f1,f2;
The length of keyword float: 5
The length of identifier f1 is: 2
The length of others , is: 1
The length of identifier f2 is: 2
The length of others ; is: 1
    int a, b, total;
The length of keyword int: 3
The length of identifier a is: 1
The length of others , is: 1
The length of identifier b is: 1
The length of others , is: 1
The length of identifier total is: 5
The length of others ; is: 1
    total = f1 + f2 + a + b;
The length of identifier total is: 5
The length of others = is: 1
The length of identifier f1 is: 2
The length of others + is: 1
The length of identifier f2 is: 2
The length of others + is: 1
The length of identifier a is: 1
The length of others + is: 1
The length of identifier b is: 1
The length of others ; is: 1
}
```

```
keywords: 3, identifiers: 11, digits: 0, others: 13,
Total number of tokens: 27
S D:\Sem-5\compiler>
```

Aim:

b. Write a LEX program to identify keywords, identifiers, numbers and other characters and generate tokens for each.

Code:

```
%option noyywrap


%{
   int c1 = 0, c2 = 0, c3 = 0, c4 = 0;
%}
digit [0-9]
letter [a-zA-Z]
%%
auto|break|case|char|const|continue|default|do|double|else|enum|extern|float|for|goto
|if|int|long|register|return|short|signed|sizeof|static|struct|switch|typedef|union|unsig
ned|void|volatile|while {printf("%s: keyword\n", yytext);c1++;}


{letter}({letter}|{digit})* {printf("%s :identifiers\n", yytext); c2++;}


{digit}+ {printf("%s :digit\n", yytext); c3++;}
```

```
. {printf("%s :others\n", yytext); c4++;}

%%

int main() {

    yylex();

    printf("keywords: %d, identifiers: %d, digits: %d ,others: %d\n", c1, c2, c3, c4);

    return 0;

}
```

Output:

```
The length of identifier total is: 5
The length of others = is: 1
The length of identifier f1 is: 2
The length of others + is: 1
The length of identifier f2 is: 2
The length of others + is: 1
The length of identifier a is: 1
The length of others + is: 1
The length of identifier b is: 1
The length of others + is: 1
The length of digit 5 is: 1
The length of others + is: 1
The length of digit 6 is: 1
The length of others + is: 1
The length of digit 7 is: 1
The length of others + is: 1
The length of digit 8 is: 1
The length of others + is: 1
The length of digit 9 is: 1
The length of others + is: 1
The length of digit 10 is: 2
The length of others ; is: 1
}
```

```
keywords: 3, identifiers: 11, digits: 6, others: 19,
Total number of tokens: 39
S D:\Sem-5\compiler>
```