

# CS6109 - COMPILER DESIGN – LAB

Week 5 – 20.09.2022

(Observations)

## Lex Program:

### 1. Reading input from a File

```
%option noyywrap
%{
    #include<stdio.h>
}%
%%

%%
main(int argc, char* argv[])
{
    if(argc > 1)
    {
        FILE *fp = fopen(argv[1], "r");
        if(fp)
            yyin = fp;
    }
    yylex();
    return 1;
}
```

### **Output:**

```
>lex filereading.l
>gcc lex.yy.c
>a.exe inp.txt
hi
hello
how are you
i am fine
```

### 2. Reading and writing into file

Replace the character with another character using file

```

%option noyywrap
%{
    #include<stdio.h>
    #include<string.h>

    char replace_with [] = "Best";
    char replace [] ="A";
%}

%%

[a-zA-Z]+    { if(strcmp(yytext, replace)==0)
                fprintf(yyout, "%s", replace_with);
                else
                fprintf(yyout, "%s", yytext);}
.            fprintf(yyout, "%s", yytext);

%%

//int yywrap() {    return 1; }
int main()
{
    extern FILE *yyin, *yyout;
    yyin=fopen("inp1.txt", "r");
    yyout=fopen("out1.txt", "w");
    yylex();
    return 0;
}

```

### **Inp1.txt**

```

A Art in the world
A price to sell your product
Classical Music is A Relaxing
A football player plays well

```

### **Output:**

```

>lex replace.l
>gcc lex.yy.c
>a.exe inp1.txt

```

### **out1.txt**

Best Art in the world  
Best price to sell your product  
Classical Music is Best Relaxing  
Best football player plays well

### 3. yyles(k) - returns the first k characters in yytext

```
%option noyywrap
%{
    #include<stdio.h>
}%
%%
[a-z]+ {
printf("\n Lower ="); ECHO;
yyles(3);
printf("\nThe word after yyles() = "); ECHO;
}
[a-zA-Z]+ {
printf("\nMixed letter is = "); ECHO;
}
%%
int main()
{
    yyles();
    return 0;
}
```

#### **Output:**

##### **concatenation two string**

```
Lower =concatenation
The word after yyles() = con
Lower =catenation
The word after yyles() = cat
Lower =enation
The word after yyles() = ena
Lower =tion
The word after yyles() = tio
Lower =n
The word after yyles() = n t
Lower =wo
The word after yyles() = wo
Lower =string
The word after yyles() = str
Lower =ing
The word after yyles() = ing
```

#### 4. yymore() - returns the next token

```
%{  
  
%}  
  
%%  
[a-z]+ {  
printf("\nLowercase letter = "); ECHO;  
printf("\nStart of 1st yymore\n");  
yymore();  
printf("\nEnd of 1st yymore\n");  
}  
  
[A-Z]+ {  
printf("\nUppercase letter = "); ECHO;  
printf("\nStart of 2nd yymore\n");  
yymore();  
printf("\nEnd of 2nd yymore\n");  
}  
%%  
main()  
{  
yylex();  
}
```

#### Output:

##### **Good EXAMple**

```
Uppercase letter = G  
Start of 2nd yymore
```

```
End of 2nd yymore
```

```
Lowercase letter = Goo  
Start of 1st yymore
```

```
End of 1st yymore
```

```
Uppercase letter = Good  
Start of 2nd yymore
```

```
End of 2nd yymore
```

Go0D  
Uppercase letter = EXAM  
Start of 2nd ymore

End of 2nd ymore

Lowercase letter = EXAMple  
Start of 1st ymore

End of 1st ymore  
EXAMple

5. Convert an infix expression to prefix expression.

\*\*\*\* Need to handle the operators +, \*, -, / including brackets

<b>Input</b>	<b>Prefix</b>
a + b * e	+a*be
a*(b+c)/d	/*a+bcd
((a+b)*c)-d	-*+abcd
(a+b)*c-(d-e)*(f+g)	-*+abc*-de+fg

**Note:**

- Consider the inputs for each program from text file.