Fokane Sakshi Anil
TE - A - 42

Page No.
Date

Assignment No: 6

LEX Program

- Aim :- Design lex program to count no. of words line & charicters of given i/p file.

- Problem Statement : Write a program using lex specification to implements lexical analysis phase of compiler to count no. of word line & character of given input file.

- Pre requisits :- LEX Basics.

- Software Requirment :

S:NO	facilities	Quantity
1	System	1
2	OIS	Ubuntu Kylin
3	S/w name	lex Tool.

- Objective :
1) To understand lex concepts
2) To implement lex program for No. of count.
3) To study about lex & java.
4) To Know important about lexical analyzer.

* How the i/p is matched :-

when the generated Scanner is run, it analyzer its inputs looking for strings which match any of its pattern. if it finds more than one match it takes the one matching the most text it finds two or more matches of the same length the rules listed first in the flex input file is choosen - once the match is determined the text. the text corresponding to the match (called the token) is made available in the global character pointer, "yytext" & its length in the global integer, "yyleng".

*Conclusion :- Thus we have studied lexical analyzer & implement an application for lexical analyzer to count total no. of word, char & line etc

Assignment No. 06

Name:- Fokane Sakshi Anil
Std-Div:-TE-A
Roll No:-42

1. Code b3.l:

```
%{
int no_line=0; int
no_space=0; int
no_char=0; int
no_words=0;
#include<string.h>
%}

%%
([a-zA-Z])+ {no_words++; no_char+=strlen(yytext);}
[" "] {no_space++;}
["\n"] {no_line++;}
. ;

%%


int yywrap(){

}

int main(int argc,char* argv[]){
yyin=fopen("test.txt","r");
   yylex();
   printf("Total Spaces %d\n",no_space);
printf("Total Words %d\n",no_words);
printf("Total Line %d\n",no_line);
no_char+=no_space;
   printf("Total Char %d\n",no_char);
```

```
    fclose(yyin);
}
```

2. text.txt File:

// *Content of text.txt File*

The earliest foundations of what would become computer science predate the invention of the modern digital computer. Machines for calculating fixed numerical tasks such as the abacus have existed since antiquity, aiding in computations such as multiplication and division. Algorithms for performing computations have existed since antiquity, even before the development of sophisticated computing equipment.

Computer science, the study of computers and computing, including their theoretical and algorithmic foundations, hardware and software, and their uses for processing information. The discipline of computer science includes the study of algorithms and data structures, computer and network design, modeling data and information processes, and artificial intelligence. Computer science draws some of its foundations from mathematics and engineering and therefore incorporates techniques from areas such as queueing theory, probability and statistics, and electronic circuit design. Computer science also makes heavy use of hypothesis testing and experimentation during the conceptualization, design, measurement, and refinement of new algorithms, information structures, and computer architectures.

OUTPUT:

```
sagar-ravan@Sagar-HP:~/SPOSL/LexProgram$ lex b3.l
sagar-ravan@Sagar-HP:~/SPOSL/LexProgram$ gcc lex.yy.c
sagar-ravan@Sagar-HP:~/SPOSL/LexProgram$ ./a.out test.txt
Total Spaces 155
Total Words 157
Total Line 3
Total Char 1180
sagar-ravan@Sagar-HP:~/SPOSL/LexProgram$
```