Fokane Sakshi Anil
TE - A - 42
Assignment NO:- 5

LEX Program

- Aim:- Design lex program for to generate token of given input file.

- Problem Statement :- write a program using lex specification to implement lexical analysis phase of compiler to generate tokens of subset of java prog.

- prerequisites : lex 110, lex 120, lex 130, lex 140, lex 160, 250.

- Objectives :- 1) To understand LEX concepts.
  2) To implement LEX program
  3) To study about LEX & Java.
  4) To know important about lexical analyer.

* Theory :-
LEX Stands for lexical Analyzer. lex is tool for generating Scanners. Scanner are programs that recognize lexical pattern in text. These lexical pattern are defined in a particular. Syntax A matched regular expression may have an associated action this action may also include returning a token when lex receives input in the form of a file or text, it takes input one

character at a time and continues until a pattern is matched, then lex perform the associated action. If on the lex & Yacc highly coupled.

- Regular Expression in lex :-

A Regular Expression is a pattern description using a meta language. An Expression is made up of Symbols. Normal Symbol are characters & no. but there are other Symbols that have Special meaning in lex. The following two tables define some of the Symbol used in lex give a few typical examples.

- Programming in lex :-

programming in lex can be divided into three Step 1) specify the pattern 2) Run lex over this file to generate C code for the Scanner. 3) Compile & link the C code to produce the executable scanner. Note : If the Scanner is part of a parser developed using Yacc. Only Step 1 & 2 Should be performed. main() for example would typically be founding the third Section. These Section are delimited by %% so to get back to the word to word counting lex prog.

Two operator Supported in a character class are the hyphen ("-") &

circumflex ("^"). when used between two character the hypher represent a range of character the circumflex, when used as the first character negates the expression.

---- defination ----

%%

---- rules ----

%%

---- Subroutines ----

Input to lex is divided c# into three section with %%. dividing the section. This is best illustrated by example. The first example is the shortest possible lex file %%. Input is copied to o/p one character at a time. The left %%% is always required as there must always be a rule section.

Conclusion :-

Thus, we have studied lexical analyzer to perform scan the program & generates token of subset of java.

```
//Name:Fokane Sakshi Anil
// TE-A 42
//  ASSINGNMENT:GROUP_B_2

/* Problem Statement  :
Write a program using lex specifications to implement lexical analysis phase of compiler to generate tokens of subset of 'java program'.
*/

/*definition or declaration*/
%{
        #include<stdio.h>
        FILE *fp;
%}

/*Tokenization*/
Package "import".*;
classdef "class".*
inbuiltfun "System.out.println(".*");"
mainfunction "public static void main".*
Assignment [a-zA-Z]+"=".*;
Datatype "int"|"float"|"double"
object .*"=new".*

/*Rules*/
%%
{Package} {printf("Package is %s",yytext);}
{classdef} {printf("Class is %s",yytext);}
{inbuiltfun} {printf("Inbuilt Function is %s",yytext);}
{mainfunction} {printf("Main Function is %s",yytext);}
{Assignment} {printf("Assignment Statement is %s",yytext);}
{Datatype} {printf("Data Type is %s",yytext);}
{object} {printf("%s is object",yytext);}
%%

/*Main Function*/
int main(int argc,char *argv[])
{
        fp=fopen(argv[1],"r");
        yyin=fp;
        yylex();
        return 0;
```

```
}


/* ************JAVA PROGRAM************
//Java input file for lex program
import java.util.Scanner;
class Addition
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		int a,b,sum;
		System.out.println("Enter two numbers : ");
		a=sc.nextInt();
		b=sc.nextInt();

		sum=a+b;
		System.out.println("Sum = "+sum);
	}
}

****************OUTPUT****************
unix@unix-HP-280-G1-MT:~/Desktop/TEB63/Ass.6$ lex lex.l
unix@unix-HP-280-G1-MT:~/Desktop/TEB63/Ass.6$ gcc lex.yy.c -ll
unix@unix-HP-280-G1-MT:~/Desktop/TEB63/Ass.6$ ./a.out Addition.java

//Java input file for lex program
Package is import java.util.Scanner;
Class is class Addition
{
	Main Function is public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in); is object
		Data Type is int a,b,sum;
		Inbuilt Function is System.out.println("Enter two numbers : ");
		Assignment Statement is a=sc.nextInt();
		Assignment Statement is b=sc.nextInt();
		Assignment Statement is sum=a+b;
		Inbuilt Function is System.out.println("Sum = "+sum);
	}
}*/
```