Name : Abhishek Padalkar

Class : TE Comps

Roll No. : 41

Exp. 3 : Lexical Analyzer using Lex Tool

CODE :

digits [0-9]

ltr [a-zA-Z]

alphanum [a-zA-Z0-9]

%%

("abstract"|"assert"|"boolean"|"break"|"byte"|"case"|"case"|"catch"|"char"|"class"|"const"|"default"|"do"|"double"|"else"|"enum"|"extends"|"false"|"final"|"finally"|"float"|"for"|"goto"|"if"|"implements"|"import"|"instanceof"|"int"|"interface"|"long"|"native"|"new"|"null"|"package"|"private"|"protected"|"public"|"return"|"short"|"static"|"strictfp"|"super"|"switch"|"synchronized"|"this"|"throw"|"throws"|"transient"|"true"|"try"|"void"|"volatile"|"while"|"continue") printf("found reserved word: ’%s’\n", yytext);

("//")({alphanum}|" ")\*|(("/\*")({alphanum}|" ")\*("\*/")) printf("found Comment : '%s'\n", yytext);

{digits}+ printf("found number: ’%s’\n", yytext);

{ltr}(\_|{alphanum})\* printf("found identifer: ’%s’\n", yytext);

("{"|"}"|"("|")"|";"|",") printf("found Separator : '%s'\n", yytext);

("++"|"--")({ltr}+)|({ltr}+)("++"|"--") printf("found Unary Operator : '%s'\n", yytext);

("\*"|"/"|"%"|"+"|"-") printf("found Arithmetic Operator : '%s'\n", yytext);

("<<"|">>"|">>>") printf("found Shift Operator : '%s'\n", yytext);

("<"|">"|"<="|">=") printf("found Comparison Operator : '%s'\n", yytext);

("=="|"!=") printf("found Equality Operator : '%s'\n", yytext);

("&"|"|"|"^") printf("found Bitwise Operator : '%s'\n", yytext);

("&&"|"||") printf("found Logical Operator : '%s'\n", yytext);

("? :") printf("found Ternary Operator : '%s'\n", yytext);

("="|"+="|"-="|"/="|"%="|"\*="|"&="|"^="|"|="|"<<="|">>="|">>>=") printf("found Assignment Operator : '%s'\n", yytext);

’.’ printf("found character: {%s}\n", yytext);

%%

int main(int argc, char \*\*argv)

{

printf("INPUT : ");

yylex();

}

OUTPUT :

Abhisheks-MacBook-Pro:Desktop abhishekpadalkar$ flex flex.l

Abhisheks-MacBook-Pro:Desktop abhishekpadalkar$ gcc lex.yy.c -ll

Abhisheks-MacBook-Pro:Desktop abhishekpadalkar$ ./a.out

INPUT : abhi

found identifer: ’abhi’

//This is a Comment

found Comment : '//This is a Comment'

/\*This is also a Comment\*/

found Comment : '/\*This is also a Comment\*/'

12345

found number: ’12345’

532hkeuwhfAFASF

found number: ’532’

found identifer: ’hkeuwhfAFASF’

huh54hkuo

found identifer: ’huh54hkuo’

{abhi}

found Separator : '{'

found identifer: ’abhi’

found Separator : '}'

a++

found Unary Operator : 'a++'

--b

found Unary Operator : '--b'

a<=b

found identifer: ’a’

found Comparison Operator : '<='

found identifer: ’b’

<<

found Shift Operator : '<<'

a>b

found identifer: ’a’

found Comparison Operator : '>'

found identifer: ’b’

a==b

found identifer: ’a’

found Equality Operator : '=='

found identifer: ’b’

a&b

found identifer: ’a’

found Bitwise Operator : '&'

found identifer: ’b’

a||b

found identifer: ’a’

found Logical Operator : '||'

found identifer: ’b’

a=6

found identifer: ’a’

found Assignment Operator : '='

found number: ’6’

abstract

found reserved word: ’abstract’

boid

found identifer: ’boid’

void

found reserved word: ’void’

int

found reserved word: ’int’

true

found reserved word: ’true’

a+b=c;

found identifer: ’a’

found Arithmetic Operator : '+'

found identifer: ’b’

found Assignment Operator : '='

found identifer: ’c’

found Separator : ';'

public static void main(String args[]) { int a,b,c; a=10; b=5; a+b=c; }

found reserved word: ’public’

found reserved word: ’static’

found reserved word: ’void’

found identifer: ’main’

found Separator : '('

found identifer: ’String’

found identifer: ’args’

[]found Separator : ')'

found Separator : '{'

found reserved word: ’int’

found identifer: ’a’

found Separator : ','

found identifer: ’b’

found Separator : ','

found identifer: ’c’

found Separator : ';'

found identifer: ’a’

found Assignment Operator : '='

found number: ’10’

found Separator : ';'

found identifer: ’b’

found Assignment Operator : '='

found number: ’5’

found Separator : ';'

found identifer: ’a’

found Arithmetic Operator : '+'

found identifer: ’b’

found Assignment Operator : '='

found identifer: ’c’

found Separator : ';'

found Separator : '}'