/\* program name is lexp.lex \*/

%{

%}

identifier [a-zA-Z][a-zA-Z0-9]\*

number [0-9]+

%%

int {printf("\n\t%s is a KEYWORD",yytext);}

float {printf("\n\t%s is a KEYWORD",yytext);}

void {printf("\n\t%s is a KEYWORD",yytext);}

char {printf("\n\t%s is a KEYWORD",yytext);}

{identifier} {printf("\n %s IDENTIFIER",yytext);}

{number} { printf("\n\t%s is a NUMBER",yytext);}

"= " {printf("\n\t%s is an ASSIGNMENT OPERATOR",yytext);}

"; " {printf("\n\t%s is a DELIMETER",yytext);}

"(" {printf("\n\t%s is a OPEN PARENTHESIS",yytext);}

")" {printf("\n\t%s is a CLOSE PARENTHESIS ",yytext);}

"{" {printf("\n\t%s is a OPEN BRACE ",yytext);}

"}" {printf("\n\t%s is a CLOSED BRACE",yytext);}

%%

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

{

FILE \*fp;

fp = fopen(argv[1],"r");

if(fp == NULL)

{

printf("could not open %s \n",argv[1]);

exit(0);

}

yyin = fp;

yylex();

}

int yywrap()

{

return 0;

}

**Process for Compile and Run**

flex lexp.lex

gcc lex.yy.c

./a.out sample.c