**Lex and Yacc**

**Calc1.yacc**

%{

#include <stdlib.h>

#include <stdio.h>

int yylex(void);

#include "y.tab.h"

%}

%token INTEGER

%%

program:line program|

line

line: expr '\n' { printf("%d\n",$1); }| 'n'

expr: expr '+' mulex { $$ = $1 + $3; }|

expr '-' mulex { $$ = $1 - $3; }|

mulex { $$ = $1; }

mulex: mulex '\*' term { $$ = $1 \* $3; }|

mulex '/' term { $$ = $1 / $3; }|

term { $$ = $1; }

term: '(' expr ')' { $$ = $2; }|

INTEGER { $$ = $1; }

%%

void yyerror(char \*s)

{

fprintf(stderr,"%s\n",s);

return;

}

int main(void)

{

/\*yydebug=1;\*/

yyparse();

return 0;}

yywrap()

{

return(1);

}

**Calc1.lex**

%{

#include <stdlib.h>

#include <stdio.h>

#include "y.tab.h"

void yyerror(char\*);

extern int yylval;

%}

%%

[ \t]+ ;

[0-9]+ {yylval = atoi(yytext);

return INTEGER;}

[-+\*/] {return \*yytext;}

"(" {return \*yytext;}

")" {return \*yytext;}

\n {return \*yytext;}

. {char msg[25];

sprintf(msg,"%s <%s>","invalid character",yytext);

yyerror(msg);

}

%%

**OUTPUT**

