A4.l

%{

#include"y.tab.h"

extern int yylval;

%}

%%

for { return (FOR); }

"(" { return (OPBR); }

")" { return (CLBR); }

";" {return (SEMIC); }

"=" { return (EQU); }

"<"|">"|"<="|">="|"^=" { return (RELOP); }

"++" { return (INC); }

"--" { return (DEC); }

[a-zA-Z]+ {yylval=yytext[0];return(ID);}

[0-9]+ {yylval=atof(yytext);return(NUM);}

%%

%{

#include<stdio.h>

int flag=0;

%}

%token FOR OPBR CLBR SEMIC RELOP EQU ID NUM INC DEC RELOP RELOP RELOP RELOP

%%

S:FOR OPBR E1 SEMIC E2 SEMIC E3 CLBR {printf

("Accepted!");flag=1;}

;

E1:ID EQU ID

| ID EQU NUM

;

E2:ID RELOP ID

| ID RELOP NUM

;

E3:ID INC

| ID DEC

;

%%

main()

{

yyparse();

}

yyerror()

{

if(flag==0)

printf("For Loop Error");

}

yywrap()

{

return 1;

}

Output :

/\*

sammyak@sammyak-Inspiron-3542:~$ lex a4.l

sammyak@sammyak-Inspiron-3542:~$ yacc -d a4.y

sammyak@sammyak-Inspiron-3542:~$ cc lex.yy.c y.tab.c -w

sammyak@sammyak-Inspiron-3542:~$ ./a.out

for(i=0;i<5;i++)

Accepted!

sammyak@sammyak-Inspiron-3542:~$ ./a.out

sammyak@sammyak-Inspiron-3542:~$ ./a.out

if(c < 0)

For Loop Errorsammyak@sammyak-Inspiron-3542:~$\*/