LEX :

%{

#include "y.tab.h"

%}

%{

yylval = 0; /\* do this for every token not using yylval, to be on the safe side \*/

%}

%%

[0-9]+ {yylval = (int)yytext; return NUMBER;}

/\* cast pointer to int for compiler warning \*/

[ \t\n] ;

"+" return(PLUS);

"-" return(MINUS);

"\*" return(TIMES);

"/" return(DIVIDE);

"^" return(POWER);

"(" return(LEFT\_PARENTHESIS);

")" return(RIGHT\_PARENTHESIS);

";" return(END);

%%

int yywrap (void) {

return 1; }

YACC :

%{

#include <stdio.h>

typedef struct node

{

struct node \*left;

struct node \*right;

char \*token;

} node;

node \*mknode(node \*left, node \*right, char \*token);

void printtree(node \*tree);

#define YYSTYPE struct node \*

#define YYSTYPE int

%}

%start lines

%token NUMBER

%token PLUS MINUS TIMES DIVIDE POWER

%token LEFT\_PARENTHESIS RIGHT\_PARENTHESIS

%token END

%left PLUS MINUS

%left TIMES DIVIDE

%right POWER

%%

lines: /\* empty \*/

| lines line /\* do nothing \*/

line: exp END { printtree($1); printf("\n");}

;

exp : term {$$ = $1;}

| exp PLUS term {$$ = mknode($1, $3, "+");}

| exp MINUS term {$$ = mknode($1, $3, "-");}

;

term : factor {$$ = $1;}

| term TIMES factor {$$ = mknode($1, $3, "\*");}

;

factor : NUMBER {$$ = mknode(0,0,(char \*)yylval);}

| LEFT\_PARENTHESIS exp RIGHT\_PARENTHESIS {$$ = $2;}

;

%%

int main (void) {return yyparse ( );}

node \*mknode(node \*left, node \*right, char \*token)

{

/\* malloc the node \*/

node \*newnode = (node \*)malloc(sizeof(node));

char \*newstr = (char \*)malloc(strlen(token)+1);

strcpy(newstr, token);

newnode->left = left;

newnode->right = right;

newnode->token = newstr;

return(newnode);

}

void printtree(node \*tree)

{

int i;

if (tree->left || tree->right)

printf("(");

printf(" %s ", tree->token);

if (tree->left)

printtree(tree->left);

if (tree->right)

printtree(tree->right);

if (tree->left || tree->right)

printf(")");

}

int yyerror (char \*s) {fprintf (stderr, "%s\n", s);}

OUTPUT :

sammyak@sammyak-Inspiron-3542:~$ lex cl1b6.l

sammyak@sammyak-Inspiron-3542:~$ yacc -d cl1b6.y

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

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

4+(5\*6+(6-4))+4;

( + ( + 4 ( + ( \* 5 6 )( - 6 4 ))) 4 )