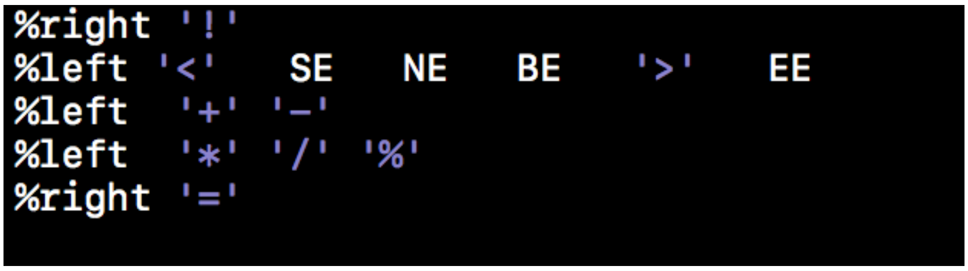
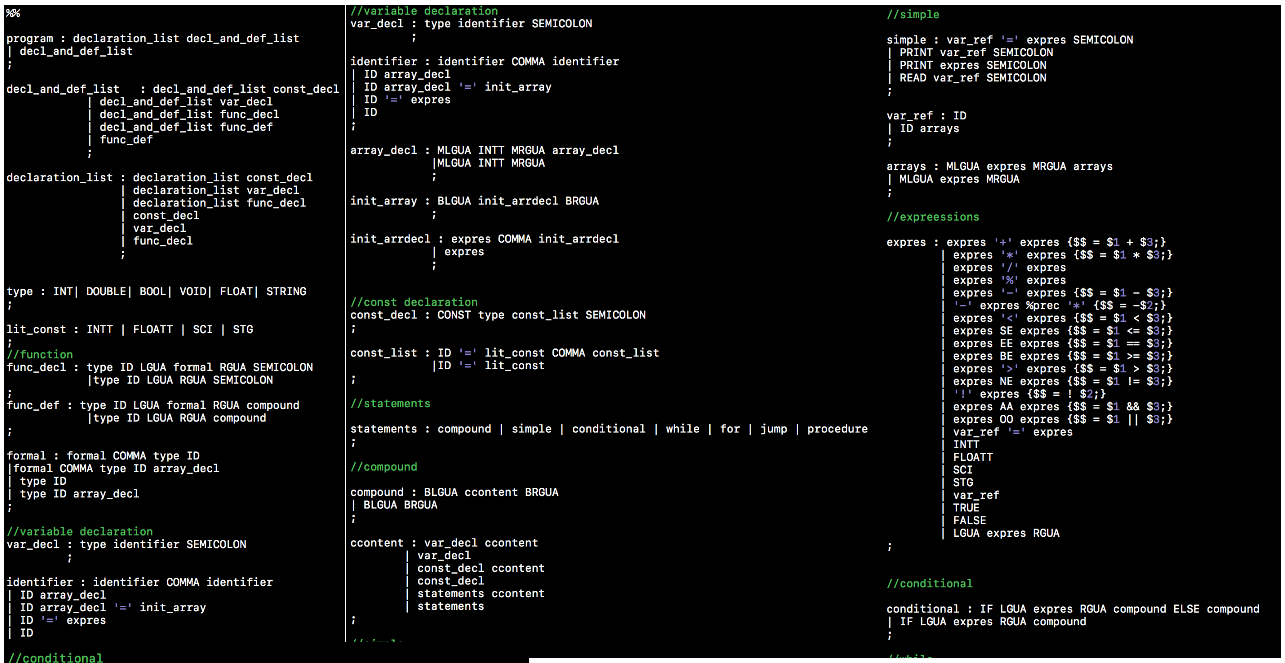


```
%right '!'
%left '<'    SE    NE    BE    '>'    EE
%left   '+'  '-'
%left   '*'  '/'  '%'
%right  '='
```

left recursion of the expression

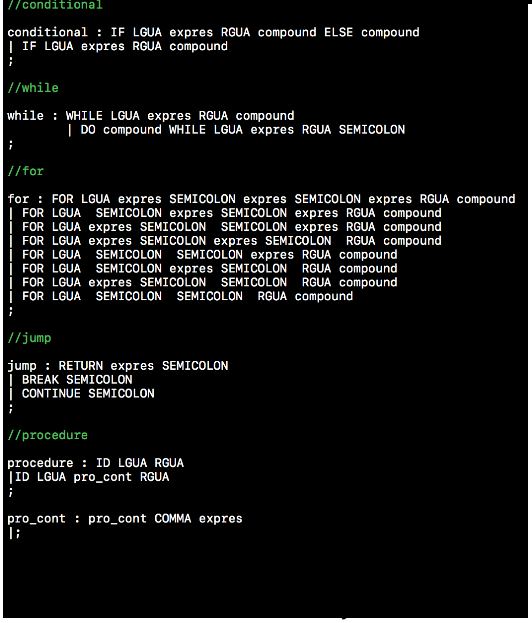
the grammer



```
%%

program : declaration_list decl_and_def_list
| decl_and_def_list
;

decl_and_def_list   : decl_and_def_list const_decl
            | decl_and_def_list var_decl
            | decl_and_def_list func_decl
            | decl_and_def_list func_def
            | func_def
            ;

declaration_list : declaration_list const_decl
                 | declaration_list var_decl
                 | declaration_list func_decl
                 | const_decl
                 | var_decl
                 | func_decl
                 ;


type : INT| DOUBLE| BOOL| VOID| FLOAT| STRING
;

lit_const : INTT | FLOATT | SCI | STG
;
//function
func_decl : type ID LGUA formal RGUA SEMICOLON
            |type ID LGUA RGUA SEMICOLON
;
func_def : type ID LGUA formal RGUA compound
            |type ID LGUA RGUA compound
;

formal : formal COMMA type ID
|formal COMMA type ID array_decl
| type ID
| type ID array_decl
;

//variable declaration
var_decl : type identifier SEMICOLON
        ;

identifier : identifier COMMA identifier
| ID array_decl
| ID array_decl '=' init_array
| ID '=' expres
| ID
```

```
//variable declaration
var_decl : type identifier SEMICOLON
        ;

identifier : identifier COMMA identifier
| ID array_decl
| ID array_decl '=' init_array
| ID '=' expres
| ID
;

array_decl : MLGUA INTT MRGUA array_decl
            |MLGUA INTT MRGUA
            ;

init_array : BLGUA init_arrdecl BRGUA
            ;

init_arrdecl : expres COMMA init_arrdecl
            | expres
            ;


//const declaration
const_decl : CONST type const_list SEMICOLON
;

const_list : ID '=' lit_const COMMA const_list
            |ID '=' lit_const
;

//statements

statements : compound | simple | conditional | while | for | jump | procedure
;

//compound

compound : BLGUA ccontent BRGUA
| BLGUA BRGUA
;

ccontent : var_decl ccontent
        | var_decl
        | const_decl ccontent
        | const_decl
        | statements ccontent
        | statements
;
```

```
//simple

simple : var_ref '=' expres SEMICOLON
| PRINT var_ref SEMICOLON
| PRINT expres SEMICOLON
| READ var_ref SEMICOLON
;

var_ref : ID
| ID arrays
;

arrays : MLGUA expres MRGUA arrays
| MLGUA expres MRGUA
;

//expreessions

expres : expres '+' expres {$$ = $1 + $3;}
        | expres '*' expres {$$ = $1 * $3;}
        | expres '/' expres
        | expres '%' expres
        | expres '-' expres {$$ = $1 - $3;}
        | '-' expres %prec '*' {$$ = -$2;}
        | expres '<' expres {$$ = $1 < $3;}
        | expres SE expres {$$ = $1 <= $3;}
        | expres EE expres {$$ = $1 == $3;}
        | expres BE expres {$$ = $1 >= $3;}
        | expres '>' expres {$$ = $1 > $3;}
        | expres NE expres {$$ = $1 != $3;}
        | '!' expres {$$ = ! $2;}
        | expres AA expres {$$ = $1 && $3;}
        | expres OO expres {$$ = $1 || $3;}
        | var_ref '=' expres
        | INTT
        | FLOATT
        | SCI
        | STG
        | var_ref
        | TRUE
        | FALSE
        | LGUA expres RGUA
;


//conditional

conditional : IF LGUA expres RGUA compound ELSE compound
| IF LGUA expres RGUA compound
;
```



```
//conditional

conditional : IF LGUA expres RGUA compound ELSE compound
| IF LGUA expres RGUA compound
;

//while

while : WHILE LGUA expres RGUA compound
        | DO compound WHILE LGUA expres RGUA SEMICOLON
;

//for

for : FOR LGUA expres SEMICOLON expres SEMICOLON expres RGUA compound
| FOR LGUA  SEMICOLON expres SEMICOLON expres RGUA compound
| FOR LGUA expres SEMICOLON  SEMICOLON expres RGUA compound
| FOR LGUA expres SEMICOLON expres SEMICOLON  RGUA compound
| FOR LGUA  SEMICOLON  SEMICOLON expres RGUA compound
| FOR LGUA  SEMICOLON expres SEMICOLON  RGUA compound
| FOR LGUA expres SEMICOLON  SEMICOLON  RGUA compound
| FOR LGUA  SEMICOLON  SEMICOLON  RGUA compound
;

//jump

jump : RETURN expres SEMICOLON
| BREAK SEMICOLON
| CONTINUE SEMICOLON
;

//procedure

procedure : ID LGUA RGUA
|ID LGUA pro_cont RGUA
;

pro_cont : pro_cont COMMA expres
|;
```

```
linux1 [/u/cs/103/0316219/compiler2] -yslin0816- % make
yacc -d -v yacctemplate.y
yacctemplate.y: warning: 3 shift/reduce conflicts [-Wconflicts-sr]
lex lextemplate.l
gcc -o parser lex.yy.c y.tab.c -ly -lfl -lbsd
linux1 [/u/cs/103/0316219/compiler2] -yslin0816- % ./parser general1.cm

|--------------------------------|
|  There is no syntactic error!  |
|--------------------------------|
```