MiniC词法和语法定义如下：

program → decl\_list /\*程序由变量描述或函数描述组成（decl）\*/

decl\_list → decl\_list decl | decl

decl → var\_decl | fun\_decl

var\_decl → type\_spec IDENT ; | type\_spec IDENT [ int\_literal] ; /\*变量包括简单变量和一维数组变量\*/

type\_spec → VOID | INT /\*函数返回值类型或变量类型包括整型或VOID\*/

fun\_decl → type\_spec IDENT ( params ) compound\_stmt

params → param\_list | VOID /\*函数参数个数可为0或多个\*/

param\_list → param\_list , param | param

param → type\_spec IDENT | type\_spec IDENT [int\_literal]

stmt\_list → stmt\_list stmt | ε

stmt → expr\_stmt | compound\_stmt | if\_stmt | while\_stmt |return\_stmt | continue\_stmt|break\_stmt

expr\_stmt → IDENT = expr; | IDENT [ expr ] = expr ; | ; /\*赋值语句\*/

while\_stmt → WHILE ( expr ) stmt /\*WHILE语句\*/

compound\_stmt→ { local\_decls stmt\_list } /\*函数内部描述，包括局部变量和语句描述\*/

local\_decls → local\_decls local\_decl | ε /\*函数内部变量描述\*/

local\_decl → type\_spec IDENT ; | type\_spec IDENT [int\_literal ] ;

if\_stmt → IF ( expr ) stmt | IF ( expr ) stmt ELSE stmt

return\_stmt → RETURN ; | RETURN expr ;

expr → expr OR expr /\*逻辑或表达式，运算符为’||’\*/

→ expr EQ expr | expr NE expr /\*关系表达式\*/

→ expr LE expr | expr < expr | expr GE expr | expr > expr /\*关系表达式\*/

→ expr AND expr /\*逻辑与表达式，运算符为’&&’\*\*/

→ expr + expr | expr – expr /\*算术表达式\*/

→ expr \* expr | expr / expr | expr % expr /\*算术表达式\*/

→ ! expr | - expr | + expr|$ expr /\*$ expr为取端口地址为expr值的端口值\*/

→ ( expr )

→ IDENT | IDENT [ expr ] | IDENT ( args ) /\* IDENT ( args )为函数调用\*/

→ int\_literal /\*数值常量\*/

int\_literal → decnum /\*数值常量是十进制整数\*/

decnum →[0..9]+ /\*十进制数定义\*/

arg\_list → arg\_list , expr | expr

args → arg\_list | ε

注：（1）其中IDENT是表示标识符，标识符可以是字母开头的字母数字串

（2）ε表示为空

（3）规定嵌套的IF语句中，ELSE与最近的IF配对

（4）运算符优先级是单目运算符高于双目运算符，算术运算符高于逻辑运算符

（5）单目运算符采用右结合规则，双目运算符采用左结合规则