Program	→ decl_list main_func
decl_list	→ decl_list decl | E
decl	→ var_decl
var_decl	→ type-spec IDENT ; | type-spec IDENT[ ] ;
type_spec	→ VOID | BOOL | INT | FLOAT
main_func	→ int main ( ) { stmt_list } | int main ( ) compound_stmt
stmt_list	→ stmt_list stmt | E
stmt	→ if_stmt | while_stmt | return_stmt | expr_stmt |
	for_stmt | break_stmt | print_stmt | var_decl
expr_stmt	→ expr ; | ;
while_stmt	→ WHILE (expr) { st_list }
st_list	→ st_list st | E
st	→ if_stmt | break_stmt | expr_stmt | print_stmt | var_decl
for_stmt	→ FOR (for_expr ; for_expr ; for_expr) { st_list }
for_expr	→ expr | E
compound_stmt	→ { local_decls stmt_list }
local_decls	→ local_decls local_decl | E
local_decl	→ type-spec IDENT ; | type-spec IDENT[ ] ;
print_stmt	→ printf (STRING_LIT);
if_stmt	→ IF (expr) { st_list }
break_stmt	→ BREAK ;
return_stmt	→ RETURN ; | RETURN expr ;

The following expressions are listed in order of increasing precedence:

expr	→ IDENT = expr
	➔ expr EQ expr | expr NE expr
	➔ expr LE expr | expr<expr | expr GE expr | expr>expr
	➔ expr + expr | expr – expr
	➔ (expr)
	➔ IDENT
	➔ BOOL_LIT | INT_LIT | FLOAT_LIT |STRING_LIT