Grammar for cin-n-out

Program -> function | program function

function -> INT ID L_PAREN arguments R_PAREN L_C_BRAC stmts R_C_BRAC

arguments -> INT ID | arguments COMMA INT ID

stmts -> SEMI | stmt SEMI | stmts stmt SEMI

stmt -> val | one_dim_arr | assign_stmt | arithmetic_op_stmt

| relational_op_stmt | while_loop | do_while_loop | for_loop

| BREAK | if_else_stmt | io_stmt

val -> INT ID

one_dim_arr -> INT ID L_S_BRAC NUM R_S_BRAC | INT ID L_S_BRAC NUM

R_S_BRAC ASSIGN L_C_BRAC nums R_C_BRAC

nums -> NUM | nums COMMA NUM

assign_stmt -> INT ID ASSIGN NUM | ID ASSIGN NUM

| INT ID L_S_BRAC NUM R_S_BRAC ASSIGN NUM

| ID L_S_BRAC exp R_S_BRAC ASSIGN NUM

arithmetic_op_stmt -> ID ASSIGN add_exp

relational_op_stmt -> equal_exp

equal_exp -> less_then_exp | equal_exp EQ equal_exp

| equal_exp NEQ equal_exp

less_then_exp -> exp | less_then_exp GT less_then_exp

| less_then_exp GTE less_then_exp | less_then_exp LT less_then_exp

| less_then_exp LTE less_then_exp

while_loop -> WHILE L_PAREN relational_op_stmt R_PAREN L_C_BRAC stmts R_C_BRAC

do_while_loop -> DO L_C_BRAC stmts R_C_BRAC WHILE L_PAREN relational_op_stmt R_PAREN

for_loop -> FOR L_PAREN assign_stmt SEMI relational_op_stmt SEMI assign_stmt R_PAREN L_C_BRAC stmts R_C_BRAC

if_else_stmt -> IF L_PAREN relational_op_stmt R_PAREN L_C_BRAC stmts R_C_BRAC

| IF L_PAREN relational_op_stmt R_PAREN L_C_BRAC stmts R_C_BRAC ELSE L_C_BRAC stmts R_C_BRAC

io_stmt -> INPUT ID SEMI

| OUTPUT ID SEMI

add_exp -> mul_exp | add_exp ADD add_exp | add_exp SUB add_exp

mul_exp -> exp | mul_exp MUL mul_exp | mul_exp DIV mul_exp

| mul_exp MOD mul_exp

exp -> NUM | SUB exp | ID | L_PAREN add_exp R_PAREN

| L_PAREN relational_op_stmt R_PAREN