if\_statement: IF LP expression RP block\_stmt ( ELSE if\_statement | ELSE block\_stmt)?;  
  
// FOR STATEMENT  
for\_statement:  
FOR (  
(ID | UNDERSCORE) COMMA (ID | UNDERSCORE) SHORT\_ASSIGN RANGE expression block\_stmt // Range form  
| for\_init (SEMI | NEWLINE) expression (SEMI | NEWLINE) for\_update block\_stmt // Three-part form  
| expression block\_stmt  
);  
  
for\_init:  
ID SHORT\_ASSIGN expression // Short declaration  
| ID assign\_op expression // Assignment  
| VAR ID type\_name? ASSIGN expression; // Variable declaration  
  
for\_update: ID assign\_op expression;

element\_access: LSB expression RSB;

field\_access: DOT ID;

call\_expr: LP list\_expression? RP;

*// Statements*

declared\_statement: variables\_declared | constants\_declared;

*// Assignments:    lhs op expression*

assign\_statement: assign\_lhs assign\_op expression (SEMI | NEWLINE);

assign\_op: ASSIGN | ADD\_ASSIGN | SUB\_ASSIGN | MUL\_ASSIGN | DIV\_ASSIGN | MOD\_ASSIGN | SHORT\_ASSIGN;

assign\_lhs: ID more\_access;

more\_access: | (field\_access | element\_access) more\_access;

expr\_list: expression | expression COMMA expr\_list;

expression: expression OR expression1 | expression1;

expression1: expression1 AND expression2 | expression2;

expression2: expression2 (EQUAL | NOT\_EQUAL) expression3 | expression3;

expression3:

    expression3 (LESS | LESS\_OR\_EQUAL | GREATER | GREATER\_OR\_EQUAL) expression4

    | expression4;

expression4: expression4 (ADD | SUB) expression5 | expression5;

expression5: expression5 (MUL | DIV | MOD) expression6 | expression6;

expression6: NOT expression6 | SUB expression6 | expression7;

*//-1.c -> failed, a.b -> passed*

expression7: operand more\_access\_expr;

more\_access\_expr: | (element\_access | field\_access | call\_expr) more\_access\_expr;

*// Operands*

operand: literal | ID | LP expression RP;

typed\_array\_literal: array\_type LB literal\_list RB;

untyped\_array\_literal: LB literal\_list RB;

array\_type: LSB (INT\_LIT | ID) RSB more\_dimensions type\_name;

more\_dimensions: | LSB (INT\_LIT | ID) RSB more\_dimensions;

type\_name: INT | FLOAT | STRING | BOOLEAN | ID | array\_type;

*// Struct literal*

struct\_literal: ID LB optional\_field\_list RB;

optional\_field\_list: | field\_list;