Green represents terminals

[] 🡪 Optional

{} 🡪 zero or more times

If they exist in black color, they are part of the language

1. program 🡪 declaration-list
2. declaration-list 🡪 declaration { declaration }
3. declaration 🡪 var-declaration | stmt
4. var-declaration 🡪 type-specifier ID;
5. type-specifier 🡪 int | float
6. stmt-list 🡪 { stmt }
7. stmt 🡪 expression-stmt | selection-stmt | iteration-stmt | read-stmt | write-stmt
8. expression-stmt 🡪 [ expression ] ;
9. selection-stmt 🡪 if ( expression ) stmt [ else stmt ]
10. iteration-stmt 🡪 while ( expression ) stmt
11. expression 🡪 [let var = ] simple-expression
12. var 🡪 ID
13. simple-expression 🡪 additive-expression [ relop additive-expression ]
14. relop 🡪 <= | < | > | >= | == | !=
15. additive-expression 🡪 term { addop term }
16. addop 🡪 + | -
17. term 🡪 factor { mulop factor }
18. mulop 🡪 \* | /
19. factor 🡪 ( expression ) | var | FLOAT | INT
20. write-var 🡪 ID | STRING
21. write-stmt 🡪 write write-var
22. read-var 🡪 ID
23. read-stmt 🡪 read ID {, ID}

If HAS\_COMPOUND is enabled as a preprocessor, then the following rules are changed / added:

1. stmt 🡪 expression-stmt | selection-stmt | iteration-stmt | read-stmt | write-stmt | cmp-stmt
2. cmp-stmt 🡪 { stmt-list }