1.declaration\_list-> declaration { declaration }

2.declaration -> type\_specifier ID declaration1

3.declaration1 -> ; | [ NUM ]; | ( params ) compound\_stmt

4.type\_specifier -> int | void

5.param\_list -> param\_list1 | param\_list2

6.param\_list1-> void [ ID [ [ ] ] { , param } ]

7.param\_list2-> int ID [ [ ] ] { , param }

8.param -> type\_specifier ID [ [ ] ]

9.compound\_stmt -> { { var\_declaration } { statement } }

10. statement -> expression\_stmt | compound\_stmt | selection\_stmt | iteration\_stmt | return\_stmt

11. expression\_stmt -> expression ; | ;

12.selection\_stmt -> if ( expression ) statement [ else statement ]

13. iteration\_stmt -> while ( expression ) statement

14.return\_stmt -> return [ expression\_stmt]

15.expression ->simple\_expression [=expression]

16.simple\_expression -> additive\_expression [ relop additive\_expression ]

17.relop -> <= | < | > | >= | == | ~=

18.additive\_expression -> term { addop term }

19.addop -> + | -

20.term -> factor { mulop factor }

21.mulop -> \* | /

22.factor -> ( expression ) | var | NUM |ID (args)

23.var -> ID [ [expression] ]

24.arg\_list -> expression { , expression } | ε