1. Program -> declaration-list
2. Declaration-list -> declaration declaration-list’

Declaration-list’ -> declaration declaration-list’ | empty

1. Declaration -> type-specifier ID declaration’

Declaration’ -> Var-declaration | Fun-declaration

1. Var-declaration -> **;**  | **[ NUM ] ;**
2. Type-specifier -> **int** | **void**
3. Fun-declaration -> **(** params **)** compound-stmt
4. Params -> param-list | **void**
5. Param-list -> param param-list’

Param-list’ -> , param param-list’ | empty

1. Param -> type-specifier  **ID** param’

Param’ -> **[ ]** | empty

1. Compound-stmt -> **{** local-declarations statement-list **}**
2. Local-declarations -> var-declaration local-declarations | empty
3. Statement-list -> statement statement-list | empty
4. Statement -> expression-stmt | compound-stmt | selection-stmt | iteration-stmt | return-stmt
5. Expression-stmt -> expression **;** | **;**
6. Selection-stmt -> **if (** expression **)** statement selection-stmt’

Selection-stmt’ -> **else** statement selection-stmt’ | empty

1. Iteration-stmt -> **while (** expression **)** statement
2. Return-stmt -> **return** return-stmt’

Return-stmt’ -> ; | expression ;

1. Expression -> var **=** expression | simple-expression
2. Var -> **ID** | **ID [** expression **]**
3. Simple-expression -> additive-expression simple-expression’
4. Simple-expression’ -> relop additive-expression | empty
5. Relop -> **<=** | **<** | **>** | **>=** | **==** | **!=**
6. Additive-expression -> term additive-expression’
7. Additive-expression’ -> addop term additive-expression’ | empty
8. Addop -> **+** | **-**
9. Term -> factor term’

Term’ -> mulop factor term’ | empty

1. Mulop -> **\*** | **/**
2. Factor -> **(** expression **)** | **ID** factor’ | **NUM**

Factor’ -> **[** expression **]** | **(** args **)** | empty

1. Args -> arg-list | empty
2. Arg-list -> expression arg-list’

Arg-list’ -> , expression arg-list’ | empty