Extended BNF Grammar for C Minus

Metasymbols: The following metasymbols will be used for this grammar. { *statement* } means 0 or more repetitions of *statement*. [*statement*]$_+$ means that the statement is optional.

1. *program* → *declaration-list*
2. *declaration-list* → *declaration* { *declaration* }
3. *declaration* → *var-declaration* | *fun-declaration*
4. *var-declaration* → *type-specifier* ***ID*** [**[** ***NUM*** **]**]$_+$ **;**
5. *type-specifier* → **int** | **void**
6. *fun-declaration* →*type-specifier* ***ID*** **(** *params* **)** *compound-stmt*
7. *params* → **void** | *param-list*
8. *param-list* → *param* { **,** *param* }
9. *param* → *type-specifier* ***ID*** [**[]**]$_+$
10. *compound-stmt* → **{** *local-declarations* *statement-list* **}**
11. *local-declarations* → { *var-declarations* }
12. *statement-list* → { *statement* }
13. *statement* → *expression-stmt*
    | *compound-stmt*
    | *selection-stmt*
    | *iteration-stmt*
    | *assignment-stmt*
    | *return-stmt*
    | *read-stmt*
    | *write-stmt*
14. *expression-stmt* → *expression* **;** | **;**
15. *selection-stmt* → **if (** *expression* **)** *statement* [**else** *statement*]$_+$
16. *iteration-stmt* → **while (** *expression* **)** *statement*

17. *return-stmt* → **return** [*expression*]$_+$ **;**

18. *read-stmt* → **read** *variable* **;**

19. *write-stmt* → **write** *expression* **;**

20. *expression* → { *var* **=** } *simple-expression*

21. *var* → ***ID*** **[** [*expression*] **]**$_+$

22. *simple-expression* → *additive-expression* [*relop additive-expression*]$_+$

22. *relop* → **<=** | **<** | **>** | **>=** | **==** | **!=**

23. *additive-expression* → *term* { *addop term* }

24. *addop* → **+** | **-**

25. *term* → *factor* { *multop factor* }

26. *multop* → ***** | **/**

27. *factor* → **(** *expression* **)** | ***NUM*** | *var* | *call*

28. *call* → ***ID*** **(** *args* **)**

29. *args* → *arg-list* | *empty*

30. *arg-list* → *expression* { **,** *expression* }

Regular Expressions

1. ***ID = letter letter****

2. ***NUM = digit digit****

3. ***letter = a*** | ***b*** | … | ***z*** | ***A*** | ***B*** | … | ***Z***

4. ***digit = 0*** | ***1*** | … | ***9***