

CNF Grammar of our language

Mobin Razavi, Ali Ahmadvand, Mohammad Davoodabadi

1. $\text{program} \rightarrow \text{declaration-list}$
2. $\text{declaration-list} \rightarrow \text{declaration-list declaration} \mid \text{declaration}$
3. $\text{declaration} \rightarrow \text{var-declaration} \mid \text{fun-declaration}$
4. $\text{var-declaration} \rightarrow \text{type-spec ID}; \mid \text{type-spec ID [NUM]};$
5. $\text{type-spec} \rightarrow \text{int} \mid \text{float} \mid \text{string}$
6. $\text{fun-declaration} \rightarrow \text{type-spec ID (params) compound-stmnt}$
7. $\text{params} \rightarrow \text{param-list} \mid \epsilon$
8. $\text{param-list} \rightarrow \text{param-list, param} \mid \text{param}$
9. $\text{param} \rightarrow \text{type-spec ID} \mid \text{type-spec ID []}$
10. $\text{compound-stmnt} \rightarrow \{\text{local-dec stmt-list}\}$
11. $\text{local-dec} \rightarrow \text{local-dec var-declaration} \mid \epsilon$
12. $\text{stmt-list} \rightarrow \text{stmt-list stmt} \mid \epsilon$
13. $\text{stmt} \rightarrow \text{compound-stmnt} \mid \text{cond-stmnt} \mid \text{iter-stmnt} \mid \text{return-stmnt} \mid \text{expression-stmnt}$
14. $\text{compound-stmnt} \rightarrow \text{if (expression) stmt} \mid \text{if (expression) stmt else stmt}$
15. $\text{iter-stmnt} \rightarrow \text{while (expression) stmt}$
16. $\text{return-stmnt} \rightarrow \text{return}; \mid \text{return expression};$
17. $\text{expression-stmnt} \rightarrow \text{expression}; \mid ;$
18. $\text{expression} \rightarrow \text{var} = \text{expression} \mid \text{simple-expression}$
19. $\text{var} \rightarrow \text{ID} \mid \text{ID [expression]}$
20. $\text{simple-expression} \rightarrow \text{logical-expression rel-op logical-expression} \mid \text{logical-expression}$
21. $\text{rel-op} \rightarrow > \mid < \mid >= \mid <= \mid == \mid !=$

22. $\text{logical-expression} \rightarrow \text{logical-expression log-op additive-expression} \mid \text{additive-expression}$
23. $\text{log-op} \rightarrow \&\& \mid \mid$
24. $\text{additive-expression} \rightarrow \text{additive-expression add-op mul-expression} \mid \text{mul-expression}$
25. $\text{add-op} \rightarrow + \mid -$
26. $\text{mul-expression} \rightarrow \text{mul-expression mul-op term} \mid \text{term}$
27. $\text{mul-op} \rightarrow * \mid /$
28. $\text{term} \rightarrow (\text{expression}) \mid \text{var} \mid \text{call} \mid \text{NUM} \mid \text{FNUM} \mid \text{string-word}$
29. $\text{call} \rightarrow \text{ID (args)} \mid \text{print(print-full-args)}$
30. $\text{print-full-args} \rightarrow \text{string} \mid \text{string args}$
31. $\text{args} \rightarrow \text{arg-list} \mid \epsilon$
32. $\text{arg-list} \rightarrow \text{arg-list, expression} \mid \text{expression}$

keywords: if else int float string return while print
 Special symbols: $+ - * / \& \mid > < >= <= == != , ;$
 $\text{ID} = \text{letter} (\text{letter} + \text{digit})^*$
 $\text{NUM} = \text{digit}^+$
 $\text{FNUM} = \text{NUM.NUM}$
 $\text{string-word} = "(\text{letter} + \text{digit} + \text{whitespace})^*"$