Juan Pablo Güitrón Leal

A01019936

Diseño de compiladores

Parser

EBNF

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 🡪 param-list | **void**
8. param-list 🡪 param { **,** param}
9. param 🡪 type-specifier **ID** [ **[ ]** ]
10. compound-stmt 🡪 **{** local-declarations statement-list **}**
11. local-declarations 🡪 { var-declaration }
12. statement-list 🡪 { statement }
13. statement 🡪 expression-stmt | compound-stmt ?| selection-stmt | iteration-stmt | return-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. expression 🡪 { var **=**} simple-expression
19. var 🡪 ID [ **[** expression **]** ]
20. simple-expression 🡪 additive-expression [ relop additive-expression ]
21. relop 🡪 **<=** | **<** | **>** | **>=** | **==** | **!=**
22. additive-expression 🡪 term { addop term }
23. addop 🡪 **+** | **-**
24. term 🡪 factor { mulop factor }
25. mulop 🡪 **\*** | **/**
26. factor 🡪 **(** expression **)** | call | var | **NUM**
27. call 🡪 **ID (** args **)**
28. args 🡪 arg-list | empty
29. arg-list 🡪 expression { **,** expression }