

Syntax Analysis Grammar

Xavier

2025 年 2 月 17 日

声明

此文法仅针对样例 c++ 源文件编写。规定大写字母为首的单词为非终结符，小写字母为首的单词为终结符，'...' 表示可出现一次或多次，etc. 表示省略。

主结构

$$\begin{aligned} \text{Program} \quad \rightarrow \quad & \text{Var_decl} \mid \text{Function_decl} \\ & \mid \text{Struct_decl} \mid \text{Expression_statement} \\ & \mid \text{Return_statement} \end{aligned}$$

变量声明

$$\begin{aligned} \text{Var_decl} \quad \rightarrow \quad & \text{Struct_decl} \mid \text{Type_keyword id} ; \\ & \mid \text{Type_keyword id} = \text{Expression} ; \\ \text{Struct_decl} \quad \rightarrow \quad & \text{struct id} \{ \text{Var_decl} \cdots \} \text{id} ; \end{aligned}$$

函数声明

$$\begin{aligned} \text{Function_decl} \quad \rightarrow \quad & \text{Type_keyword id} (\text{Param}) \{ \text{Statement} \} \\ & \mid \text{Type_keyword id} (\text{Param}) ; \\ \text{Param} \quad \rightarrow \quad & \text{Type_keyword id} (\text{Type_keyword id}) * \mid \epsilon \end{aligned}$$

语句部分

Statement \rightarrow Expression_statement | Return_statement
| Block_statement
| Declaration_statement
| If_statement | While_statement

Expression_statement \rightarrow Expression ;

Return_statement \rightarrow return Expression ;

Block_statement \rightarrow {Statement }

Declaration_statement \rightarrow Var_decl

If_statement \rightarrow if (Expression) Statement [else Statement]

While_statement \rightarrow while (Expression) Statement

表达式部分（消除左递归）

Expression \rightarrow Term ((" + " | " - ")Term)*

Term \rightarrow Factor ((" * " | "/")Factor)*

Factor \rightarrow id | Literal | (Expression)

Literal \rightarrow Numeric_literal | String_literal | Char_literal | Boolean_literal

Argument \rightarrow Expression , Argument | Expression | ϵ

Operator \rightarrow + | - | * | / |
| % | < | > | == | != |

字面量部分 (Literal)

Numeric_literal → INTEGER | FLOATING

String_literal → STRING_LITERAL

Char_literal → CHARACTER

Boolean_literal → TRUE | FALSE

类型关键字与关键字区分

Type_keyword → int | float | double | char | void | bool

Keyword → if | else | for | return | struct | switch | case | etc.