Optimized C++ Subset Grammar

Xavier

2025 年 2 月 13 日

核心文法改进

类型系统（精准区分 TYPE_KEYWORD）

$$
\begin{aligned}
Type &\rightarrow TYPE\_KEYWORD\ (\text{unsigned} \mid \text{const})^* \quad /^*\ \text{匹配 } int/char \text{ 等基础类型 } ^*/ \\
&\mid \text{struct } Identifier \quad /^*\ \text{自定义结构体类型 } ^*/ \\
ArrayDecl &\rightarrow Identifier\ [\ INTEGER\ ] \quad /^*\ \text{精确匹配数组声明 } ^*/ \\
PointerDecl &\rightarrow Type\ *+ \quad /^*\ \text{指针类型声明 } ^*/
\end{aligned}
$$

变量声明（适配 TokenType）

$$
\begin{aligned}
VarDecl &\rightarrow Type\ (Identifier \mid ArrayDecl)\ (=\ PrimaryExpr)?\ ; \quad /^*\ \text{基础声明 } ^*/ \\
&\mid \text{auto } Identifier\ =\ Expression\ ; \quad /^*\ auto\ \text{推导声明 } ^*/ \\
StructDecl &\rightarrow \text{struct } Identifier\ \{(VarDecl)+\}\ ; \quad /^*\ \text{结构体精确匹配 } ^*/
\end{aligned}
$$

函数声明（强化参数类型检查）

$$FunctionDecl \rightarrow TYPE\_KEYWORD\ Identifier\ (ParameterList\ )\ \{Statement^*\}\quad /*\ 函数定$$
$$ParameterList \rightarrow (Type\ Identifier(,\ Type\ Identifier)^*)?\quad /*\ 参数必须明确类型\ */$$

表达式系统（分层消除左递归）

$$Expression \rightarrow AssignmentExpr\quad /*\ 表达式入口\ */$$
$$AssignmentExpr \rightarrow LogicalOrExpr\ (\ =\ AssignmentExpr\ )?\quad /*\ 赋值表达式\ */$$
$$LogicalOrExpr \rightarrow LogicalAndExpr\ (\ ||\ LogicalAndExpr\ )^*\quad /*\ 逻辑或\ */$$
$$LogicalAndExpr \rightarrow EqualityExpr\ (\ \&\&\ EqualityExpr\ )^*\quad /*\ 逻辑与\ */$$
$$EqualityExpr \rightarrow RelationalExpr\ ((==\ |\ !=)\ RelationalExpr\ )^*\quad /*\ 相等比较\ */$$
$$RelationalExpr \rightarrow AdditiveExpr\ ((<\ |\ >\ |\ <=\ |\ >=)\ AdditiveExpr\ )^*\quad /*\ 关系运算\ */$$
$$AdditiveExpr \rightarrow MultiplicativeExpr\ ((+\ |\ -)\ MultiplicativeExpr\ )^*\quad /*\ 加减运算\ */$$
$$MultiplicativeExpr \rightarrow PrimaryExpr\ ((*\ |\ /\ |\ \%)\ PrimaryExpr\ )^*\quad /*\ 乘除运算\ */$$
$$PrimaryExpr \rightarrow INTEGER\ |\ FLOATING\ |\ STRING\_LITERAL\ |\ CHARACTER\quad /$$
$$|\ Identifier\ |\ (\ Expression\ )\ |\ FunctionCall\quad /*\ 变量/括号/函数调用\ */$$
$$FunctionCall \rightarrow Identifier\ (ArgumentList\ )\quad /*\ 函数调用表达式\ */$$
$$ArgumentList \rightarrow (Expression(,\ Expression)^*)?$$

语句系统（适配控制流）

$$
\begin{aligned}
Statement &\to CompoundStmt \mid IfStmt \mid WhileStmt \mid ReturnStmt \mid ExprStmt \mid VarDecl \\
CompoundStmt &\to \{Statement^*\} \quad /^*\ \text{语句块}\ ^*/ \\
IfStmt &\to \text{if } (Expression)\ Statement\ (\text{else } Statement)? \quad /^*\ \text{带 } else\ \text{分支}\ ^*/ \\
WhileStmt &\to \text{while } (Expression)\ Statement \quad /^*\ \text{循环语句}\ ^*/ \\
ReturnStmt &\to \text{return } Expression\ ; \quad /^*\ \text{严格分号结尾}\ ^*/ \\
ExprStmt &\to Expression\ ; \quad /^*\ \text{表达式必须分号结尾}\ ^*/
\end{aligned}
$$