Ciências Exatas e Engenharias
Ciência da Computação
Linguagens Formais
Ricardo Vargas Dorneles
Nícolas Erciro Zanatto

Informações:

Bugs da gramática, ainda não corrigidos:

- não consegue analisar certo structs
- quando há mais de 1 função no arquivo, ex: função fat() e main(),
  não consegue analisar corretamente os dados da segunda função, caso houver apenas uma ele consegue (precisa de alguma alteração da gramática “Translation_unit”).

Gramática (baseada no https://www.lysator.liu.se/c/ANSI-C-grammar-y.html)

Declaration_specifiers -> Type_specifier

Type_specifier -> int | float | char | void | Struct_or_union_specifier

Initializer -> Expression | Initializer'
Initializer' -> { initializer_list Initializer''
Initializer'' -> , } | }

Initializer_list -> Initializer Initializer_list'
Initializer_list' -> , Initializer Initializer_list' | vazio

Init_declarator -> Declarator Init_declarator’
Init_declarator’ -> = Initializer | vazio

Init_declarator_list -> Init_declarator Init_declarator_list'
Init_declarator_list' -> , Init_declarator Init_declarator_list' | vazio

Declaration -> Declaration_specifiers Declaration‘
Declaration‘-> ; | Declaration_specifiers Init_declarator_list ;

Declaration_list -> Declaration Declaration_list'
Declaration_list' -> Declaration Declaration_list' | vazio

Declarator -> Direct_declarator

Direct_declarator -> id | (Declarator) | Direct_declarator#
Direct_declarator# -> Direct_declarator'' Direct_declarator# | λ
Direct_declarator'' -> [Direct_declarator' | (Direct_declarator'''
Direct_declarator' -> Constant_expression] |]
Direct_declarator''' -> Parameter_type_list) | Identifier_list) |)

Parameter_type_list -> Parameter_list

Parameter_list -> Parameter_declaration Parameter_list'
Parameter_list' -> , Parameter_declaration Parameter_list' | vazio

Parameter_declaration -> Declaration_specifiers Declarator | Declaration_specifiers
Abstract_declarator | Declaration_specifiers

Identifier_list -> identifier Identifier_list'
Identifier_list' -> , identifier Identifier_list' | vazio

Function_definition -> Declaration_specifiers Declarator Function_definition’
Function_definition -> Declarator Function_definition’
Function_definition’-> Declaration_list Compound_statement
Function_definition’-> Compound_statement

Statement -> Compound_statement
Statement -> Expression_statement
Statement -> Selection_statement
Statement -> Iteration_statement
Statement -> Jump_statement

Statement_list -> Statement Statement_list'
Statement_list' -> Statement Statement_list' | vazio

Compound_statement SETA { Compound_statement'
Compound_statement' SETA }
Compound_statement' SETA Statement_list Compound_statement''
Compound_statement' SETA Declaration_list Compound_statement''
Compound_statement'' SETA }
Compound_statement'' SETA Compound_statementStatOrDeclar }

Compound_statementStatOrDeclar SETA Declaration_list Compound_statementStatOrDeclar#
Compound_statementStatOrDeclar SETA Statement_list Compound_statementStatOrDeclar#
Compound_statementStatOrDeclar# SETA Compound_statementStatOrDeclar
Compound_statementStatOrDeclar# SETA VAZIO

Expression_statement -> ; | Expression ;

Selection_statement -> if (Expression) Statement Selection_statement'
Selection_statement' -> else Statement | vazio

Iteration_statement -> while (Expression) Statement
Iteration_statement -> do Statement while (Expression) ;
Iteration_statement -> for (Expression_statement Expression_statement Iteration_statement'
Iteration_statement' ->) Statement | Expression) Statement

Jump_statement -> goto identifier ;
Jump_statement -> continue ;
Jump_statement -> break ;
Jump_statement -> return ;
Jump_statement -> return Expression ;

External_declaration -> Function_definition | Declaration

Translation_unit -> External_declaration Translation_unit'
Translation_unit' -> External_declaration Translation_unit' | vazio

EExpression SETA Assignment_expression Expression#
Expression# SETA , Assignment_expression Expression#
Expression# SETA VAZIO
Assignment_expression SETA Unary_expression Assignment_operator Assignment_expression
Assignment_expression SETA Conditional_expression
Assignment_operator SETA =
Assignment_operator SETA *=
Assignment_operator SETA /=
Assignment_operator SETA %=
Assignment_operator SETA +=
Assignment_operator SETA -=
Assignment_operator SETA <<=
Assignment_operator SETA >>=

Assignment_operator SETA &=
Assignment_operator SETA ^=
Assignment_operator SETA or_assign
Unary_expression SETA Postfix_expression
Unary_expression SETA ++ Unary_expression
Unary_expression SETA -- Unary_expression
Unary_expression SETA Unary_operator Cast_expression
Unary_expression SETA sizeof Unary_expression'
Unary_expression' SETA Unary_expression
Unary_expression' SETA (Type_name)
Postfix_expression SETA Primary_expression Postfix_expression#
Postfix_expression# SETA [Expression] Postfix_expression#
Postfix_expression# SETA (Postfix_expression#'
Postfix_expression# SETA . id Postfix_expression#
Postfix_expression# SETA -> id Postfix_expression#
Postfix_expression# SETA ++ Postfix_expression#
Postfix_expression# SETA -- Postfix_expression#
Postfix_expression# SETA VAZIO
Postfix_expression#' SETA) Postfix_expression#
Postfix_expression#' SETA Argument_expression_list) Postfix_expression#
Primary_expression SETA id
Primary_expression SETA number
Primary_expression SETA string
Primary_expression SETA (Expression)
Argument_expression_list SETA Assignment_expression Argument_expression_list#
Argument_expression_list# SETA , Assignment_expression Argument_expression_list#
Argument_expression_list# SETA VAZIO
Unary_operator SETA &
Unary_operator SETA *
Unary_operator SETA +
Unary_operator SETA -
Unary_operator SETA ~
Unary_operator SETA !
Cast_expression SETA Unary_expression
Cast_expression SETA (Type_name) Cast_expression
Type_name SETA Specifier_qualifier_list Type_name'
Type_name' SETA VAZIO
Type_name' SETA Abstract_declarator
Specifier_qualifier_list SETA Type_specifier
Abstract_declarator SETA Direct_abstract_declarator
Direct_abstract_declarator SETA (Direct_abstract_declarator'
Direct_abstract_declarator SETA [Direct_abstract_declarator'
Direct_abstract_declarator' SETA] Direct_abstract_declarator#

Direct_abstract_declarator' SETA Constant_expression] Direct_abstract_declarator#
Direct_abstract_declarator' SETA Abstract_declarator) Direct_abstract_declarator#
Direct_abstract_declarator' SETA) Direct_abstract_declarator#
Direct_abstract_declarator' SETA Parameter_type_list) Direct_abstract_declarator#
Direct_abstract_declarator# SETA [Direct_abstract_declarator#'
Direct_abstract_declarator# SETA (Direct_abstract_declarator#'
Direct_abstract_declarator# SETA VAZIO
Direct_abstract_declarator#' SETA) Direct_abstract_declarator#
Direct_abstract_declarator#' SETA Parameter_type_list) Direct_abstract_declarator#
Direct_abstract_declarator#' SETA] Direct_abstract_declarator#
Direct_abstract_declarator#' SETA Constant_expression] Direct_abstract_declarator#
Constant_expression SETA Conditional_expression
Conditional_expression SETA Logical_or_expression
Logical_or_expression SETA Logical_and_expression Logical_or_expression#
Logical_or_expression# SETA or Logical_and_expression Logical_or_expression#
Logical_or_expression# SETA VAZIO
Logical_and_expression SETA Inclusive_or_expression Logical_and_expression#
Logical_and_expression# SETA && Inclusive_or_expression Logical_and_expression#
Logical_and_expression# SETA VAZIO
Inclusive_or_expression SETA Exclusive_or_expression Inclusive_or_expression#
Inclusive_or_expression# SETA bitor Exclusive_or_expression Inclusive_or_expression#
Inclusive_or_expression# SETA VAZIO
Exclusive_or_expression SETA And_expression Exclusive_or_expression#
Exclusive_or_expression# SETA ^ And_expression Exclusive_or_expression#
Exclusive_or_expression# SETA VAZIO
And_expression SETA Equality_expression And_expression#
And_expression# SETA & Equality_expression And_expression#
And_expression# SETA VAZIO
Equality_expression SETA Relational_expression Equality_expression#
Equality_expression# SETA == Relational_expression Equality_expression#
Equality_expression# SETA != Relational_expression Equality_expression#
Equality_expression# SETA VAZIO
Relational_expression SETA Shift_expression Relational_expression#
Relational_expression# SETA < Shift_expression Relational_expression#
Relational_expression# SETA > Shift_expression Relational_expression#
Relational_expression# SETA <= Shift_expression Relational_expression#
Relational_expression# SETA >= Shift_expression Relational_expression#
Relational_expression# SETA VAZIO
Shift_expression SETA Additive_expression Shift_expression#
Shift_expression# SETA << Additive_expression Shift_expression#
Shift_expression# SETA >> Additive_expression Shift_expression#
Shift_expression# SETA VAZIO
Additive_expression SETA Multiplicative_expression Additive_expression#

Additive_expression# SETA + Multiplicative_expression Additive_expression#
Additive_expression# SETA - Multiplicative_expression Additive_expression#
Additive_expression# SETA VAZIO
Multiplicative_expression SETA Cast_expression Multiplicative_expression#
Multiplicative_expression# SETA * Cast_expression Multiplicative_expression#
Multiplicative_expression# SETA / Cast_expression Multiplicative_expression#
Multiplicative_expression# SETA % Cast_expression Multiplicative_expression#
Multiplicative_expression# SETA VAZIO
Struct_or_union_specifier SETA Struct_or_union id Struct_or_union_specifier'
Struct_or_union_specifier' SETA { Struct_declaration_list }
Struct_or_union_specifier' SETA VAZIO
Struct_or_union SETA struct
Struct_or_union SETA union
Struct_declaration_list SETA Struct_declaration Struct_declaration_list#
Struct_declaration_list# SETA Struct_declaration Struct_declaration_list#
Struct_declaration_list# SETA VAZIO
Struct_declaration SETA Specifier_qualifier_list Struct_declarator_list ;
Struct_declarator_list SETA Struct_declarator Struct_declarator_list#
Struct_declarator_list# SETA , Struct_declarator Struct_declarator_list#
Struct_declarator_list# SETA VAZIO
Struct_declarator SETA Declarator Struct_declarator'
Struct_declarator SETA : Constant_expression
Struct_declarator' SETA VAZIO
Struct_declarator' SETA : Constant_expression