<team 21 Compiler assignment(simple C) 제출물>

소프트웨어학부 20214184 김재오

소프트웨어학부 20216064 양시훈

1) Context Free Grammar(CFG)

CODE' -> CODE | FDECL CODE | VDECL CODE | epsilon

VDECL -> vtype id semi | vtype ASSIGN semi

ASSIGN -> id assign RHS

RHS -> EXPR | literal | character | boolstr

EXPR -> EXPR addsub TERM | TERM | lparen EXPR rparen

TERM -> TERM multdiv FACTOR | FACTOR

FACTOR -> id | num

FDECL -> vtype id lparen ARG rparen lbrace BLOCK RETURN rbrace

ARG -> vtype id MOREARGS | epsilon

MOREARGS -> comma vtype id MOREARGS | epsilon

BLOCK -> STMT BLOCK | epsilon

STMT -> VDECL | ASSIGN semi

STMT -> if lparen COND rparen lbrace BLOCK rbrace ELSE

STMT -> while lparen COND rparen lbrace BLOCK rbrace

COND -> COND comp boolstr | boolstr

ELSE -> else lbrace BLOCK rbrace | epsilon

RETURN -> return RHS semi

파란 색: 수정한 CFG의 부분

2) SLR parsing table

A white sheet with many numbers

Description automatically generated with medium confidenceA white grid with blue and green dots

Description automatically generated

<How to make unambiguous CFG?>

1. EXPR -> EXPR addsub EXPR | EXPR multdiv EXPR에서 ambiguous 발생

TERM, FACTOR를 선언하여 multdiv가 addsub보다 먼저 계산 되도록 하고, 한쪽으로만 recursion이 발생하도록 rightmost nonterminal을 EXPR에서 TERM으로, TERM에서 FACTOR로 derivation 되도록 설정하였다.

2. COND -> COND comp COND에서 ambiguous 발생

왼쪽으로만 branching하도록 하기 위해, rightmost COND를 COND 대신 boolstr로 설정.

COND자리에는 결국에 boolstr만 올 수 있기 때문에, 의미적으로 동일하다.