Comp 442 A2

Alexander De Laurentiis

02/20/22

1 ll(1) grammar, Slightly codified

start –¿ program.
program –¿ structOrImplOrFuncplus.
structOrImplOrFuncplus –¿ structOrImplOrFunc, structOrImplOrFuncplus —[].
structOrImplOrFunc –¿ structDecl — implDef — funcDef.
structDecl –¿ ["struct"], id, optstructdecl2, extraId, [""], visibilityMemberDef, [""],[";"].
optstructdecl2 –¿ ["inherits"],id,[','],id —[].
implDef –¿ ["impl"],["id"], [""], funcDefPlus, [""].
funcDefPlus –¿ funcDef,funcDefPlus. funcDefPlus –¿ [].
funcDef –¿ funcHead, funcBody.
extraId –¿ id,[","],extraId — [].
visibilityMemberDecl –¿ visibilityMemberDecl,visibility, memberDecl. visibilityMemberDecl –¿ [].
visibility –¿ ["public"]. visibility –¿ ["private"].
memberDecl –¿ funcDecl. memberDecl –¿ varDecl.
funcDecl –¿ funcHead, [";"].
funcHead –¿ ["func"], ["id"],["("],fParams,[")"],["-¿"],returnType.
funcBody –¿ [""], varDeclOrStatPlus , [""].
varDeclOrStatPlus –¿ varDeclOrStat, varDeclOrStatPlus —[].
varDeclOrStat –¿ varDecl —statement.
varDecl –¿ ["let"], id(X), [":"],type,arraySizePlus,[";"].
arraySizePlus –¿ arraySize, arraySizePlus —[].
statement –¿ idnest, statement2, [";"] —["if"], ["("], relExpr, [")"], ["then"], statBlock, ["else"], statBlock, [";"] —["while"], ["("],relExpr, [")"], statBlock,[";"] —["read"], ["("], variable, [")"], [";"] —["write"], ["("], expr, [")"], [";"] —["return"], ["("], expr, [")"], [";"].
statement2 –¿ id, indice, assignop, expr — ["("], aparams, [")"].
statBlock –¿ [""], statementPlus, [""] —statement —[].
statementPlus –¿ statement, statementPlus — [].
expr –¿ arithExpr, expr2.
expr2 –¿ relOp, arithExpr —[].

arithExpr –¿ term, rightRightRecArithExpr.
rightRightRecArithExpr –¿ addop, term, rightRightRecArithExpr —[].
sign –¿ ["+"]. sign –¿ ["-"].
term –¿ factor, rightRecTerm.
rightRecTerm –¿ multop, term, rightRecTerm —[].
factor –¿ idnest, fidnest —["intLit"] —["floatLit"] —["("], arithExpr, [")"] —["not"], factor —sign, factor.
fidnest –¿ id, fid1.
fid1 –¿ indice —["("], aparams, [")"].
variable –¿ idnestPlus, ["id"], indicePlus.
functionCall –¿ idnestPlus, ["id"], ["("], aParams, [")"].
idnestPlus –¿ idnest, idnestPlus —[].
indicePlus –¿ indice, indicePlus —[]. indice –¿ ["["], arithExpr, ["]"].
idnest –¿ id, fid, ["."] —["("], aparams, [")"],["."].
arraySize –¿ ["["], arraySizeP, ["]"].
arraySizeP –¿ integer(X) —[].
type –¿ ["integer"] —["float"] —["id"].
returnType –¿ type. returnType –¿ ["void"],write("void").
fParams –¿ write("fParams"), id, [":"], type, arraySizePlus, fParamsTailPlus —[].
arraySizePlus –¿ arraySize, arraySizePlus —[].
fParamsTailPlus –¿ write("fParamsTailPlus"),fParamsTail, fParamsTailPlus —[].
fParamsTail –¿write("fParamsTail"), [","], id, [":"], type, arraySizePlus.
aParams –¿ expr, aParamsTailPlus —[].
aParamsTailPlus –¿ aParamsTail, aParamsTailPlus —[].
aParamsTail –¿ [","], expr.
assignOp –¿ ["="].
relOp –¿ ["eq"]. relOp –¿ ["neq"]. relOp –¿ ["lt"]. relOp –¿ ["gt"]. relOp –¿ ["leq"]. relOp –¿ ["geq"].
addOp –¿ ["+"]. addOp –¿ ["-"]. addOp –¿ ["or"].
multOp –¿ ["*"]. multOp –¿ ["/"]. multOp –¿ ["and"].

2 Original UCalgary ll(1)

START -¿ STRUCTORIMPLORFUNC .
ADDOP -¿ plus —minus —or .
APARAMS -¿ EXPR APARAMSTAIL —.
APARAMSTAIL -¿ comma EXPR .
ARITHEXPR -¿ TERM RIGHTRECARITHEXPR .
ARRAYSIZE -¿ lsqbr ARRAYSIZEP rsqbr .
ARRAYSIZEP -¿ intnum —.
ASSIGNOP -¿ equal .
EXPR -¿ ARITHEXPR EXPR2.
EXPR2 -¿ RELOP ARITHEXPR —.

FACTOR -¿ IDNEST FIDNEST —intlit —floatlit —lpar ARITHEXPR rpar —not FACTOR —SIGN FACTOR .

FIDNEST -¿ id Fid_1 .

Fid_1 -¿ INDICE —lpar APARAMS rpar .

FPARAMS -¿ id colon TYPE ARRAYSIZE FPARAMSTAIL —.

FPARAMSTAIL -¿ comma id colon TYPE ARRAYSIZE .

FUNCBODY -¿ lcurbr VARDECLORSTAT rcurbr .

FUNCDECL -¿ FUNCHEAD semi .

FUNCDEF -¿ FUNCHEAD FUNCBODY .

FUNCHEAD -¿ func id lpar FPARAMS rpar minusarrow RETURNTYPE .

IDNEST -¿ id Fid .

Fid -¿ INDICE dot —lpar APARAMS rpar dot .

IMPLDEF -¿ impl id lcurbr FUNCDEF rcurbr .

INDICE -¿ lsqbr ARITHEXPR rsqbr .

MEMBERDECL -¿ FUNCDECL —VARDECL .

MULTOP -¿ mult —div —and .

OPTSTRUCTDECL2 -¿ inherits id comma id —.

RELEXPR -¿ ARITHEXPR RELOP ARITHEXPR .

RELOP -¿ eq —neq —lt —gt —leq —geq .

RETURNTYPE -¿ TYPE —void .

RIGHTRECARITHEXPR -¿ —ADDOP TERM RIGHTRECARITHEXPR .

RIGHTRECTERM -¿ —MULTOP FACTOR RIGHTRECTERM .

SIGN -¿ plus —minus .

STATBLOCK -¿ lcurbr STATEMENT rcurbr —STATEMENT —.

STATEMENT -¿ IDNEST STATEMENT2 semi —if lpar RELEXPR rpar then STATBLOCK else STATBLOCK semi —while lpar RELEXPR rpar STATBLOCK semi —read lpar VARIABLE rpar semi —write lpar EXPR rpar semi —return lpar EXPR rpar semi.

STATEMENT2 -¿ id INDICE ASSIGNOP EXPR — lpar APARAMS rpar .

STRUCTDECL -¿ struct id OPTSTRUCTDECL2 lcurbr VISIBILITY MEMBERDECL rcurbr semi .

STRUCTORIMPLORFUNC -¿ STRUCTDECL —IMPLDEF —FUNCDEF .

TERM -¿ FACTOR RIGHTRECTERM .

TYPE -¿ integer —float —id .

VARDECL -¿ let id colon TYPE ARRAYSIZE semi .

VARDECLORSTAT -¿ VARDECL —STATEMENT .

VARIABLE -¿ IDNEST id INDICE.

VISIBILITY -¿ public —private .