Păucean Amalia-Crucița MIE 831

LAB 4 – LFCD

Starting from the BNF description of the minilanguage syntax, construct the context free grammar (cfg) for parsing.

G = (N, ∑, P, S)

N = { program; libList; librarie; nmsp; decList; declare; type; stmtList; stmt; simpleStmt; assignment; expression; term; factor; iostmt; structCtmt; ifStmt; whileStmt; forStmt; condition; RELATION; complexStmt } -nonterminals

∑= { int; float; double; char; string; bool; break; struct; do; if; else; while; for; true; false; void; main; cin; cout; +; -; *; /; =; <; >; <=; >=; <<; >>; ==; (;); {; }; ;; space; main; return; 0; #; include; iostream; using; namespace; std; IDENTIFIER} -terminals

P = { program -> libList nmsp int main ( ) { decList stmtList return 0;}

libList -> librarie | librarielibList

librarie -> # include < iostream >

nmsp -> using namespace std ;

decList -> declare | declare ; decList

declare -> type IDENTIFIER ;

type -> int | float | double | char | string | struct | bool

stmtList -> stmt | stmt ; stmtList

stmt -> simpleStmt | structStmt | complexStmt

simpleStmt -> assignment | iostmt

assignment -> IDENTIFIER -> expression ;

expression -> expression + term | expression - term | term

term -> term * factor | factor

factor -> IDENTIFIER | (expression)

iostmt -> cin IDENTIFIER ;| cout IDENTIFIER ;

structStmt -> ifStmt | whileStmt | forStmt

ifStmt -> if (condition) { stmtList } else { stmtList }

whileStmt -> while (condition){ stmtList }

forStmt -> for (assignment ; condition ; assignment) { stmtlist }

condition -> expression RELATION expression

RELATION -> < | > | = | == | => | =>

complexStmt -> struct IDENTIFIER { decList } ; }

S : P