Curtin University

Programming Languages

Semester 2, 2016

Assignment

Due Date: 14th October, 2016

Objective

The objective of this assignment is to gain some practical experience in writing a language specification using EBNF and in writing parsers and syntax checkers using YACC (GNU Bison) and Lex (GNU Flex).

Task 1 -EBNF specification

You are to derive the EBNF notation for the PL2016 language. The syntax graphs describing the language are included in the Appendix of this document. Your specification must be clear and consistent (use the same set of extensions throughout your solution).

Task 2 – Syntax checker

You are to design, using YACC and Lex, a parser and syntax checker named *PL2016_check* for legal PL2016 programs.

You will need to parse the input stream of characters into reserved words, numbers and identifiers using Lex. These symbols will be passed to YACC which will check to see if the sequence of tokens fits the PL2016 language.

Study the syntax graphs supplied and the BNF you have derived and convert the rules to YACC rules. A correct PL2016 program should parse without any syntax errors. If there is an error YACC outputs "syntax error". Therefore, for an incorrect PL2016 program this error will occur as soon as the syntax is detected to be incorrect. To enable readability and understanding, display all the symbols to the screen as the PL2016 code is being parsed.

Requirements

You are required to submit the following items for assessment:

1. The typed EBNF description of PL2016 derived from the syntax graphs provided.
2. A typed report that contains a description of how you built the parser and syntax checker for the PL2016 language. The report must be presented in a professional manner.
3. Fully documented YACC and Lex source code.

The final executable is to be named *PL2016_check* and must run on the Linux lab machines. You are to also specify how to build *PL2016_check*.

The command line arguments to *PL2016_check* are as follows:

PL2016_check < *PL2016_program_file* (where the PL2016_program_file is user specified).

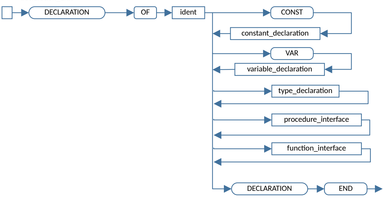
The Lex and YACC fully documented code, the executable file named *PL2016_check* and the instructions on how to compile the source files must be placed in a directory called ~/proglang/assignment (where ~ is your home directory).

The hard copy of the report should be handed in via the office in the Computer Science building before 9:00am on the due date. The electronic files containing your source code and documentation must be submitted via blackboard by 9.00am on the date specified. Please ensure that your spelling of the file names is correct!

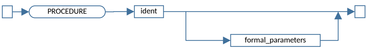
To protect against plagiarism, you must ensure that all files and directories are adequately protected

basic_program

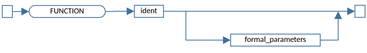
declaration_unit



procedure_interface



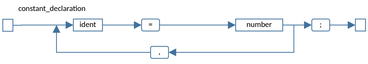
function_interface



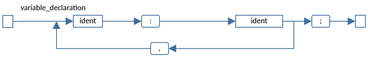
type_declaration

formal_parameters

constant_declaration



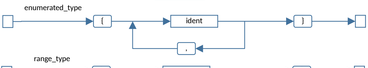
variable_declaration



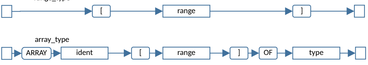
type

basic_type

enumerated_type

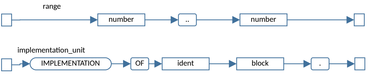


range_type



array_type

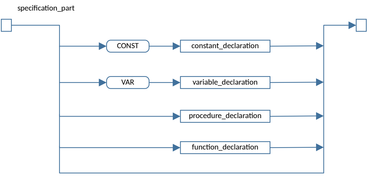
range



implementation_unit

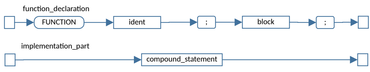
block

specification_part



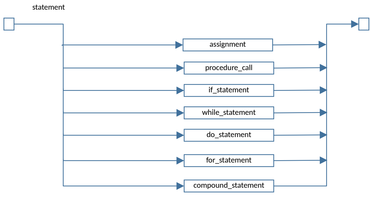
procedure_declaration

function_declaration



implementation_part

statement



assignment

procedure_call

if_statement

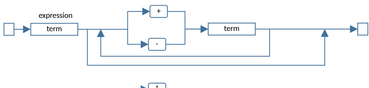
while_statement

do_statement

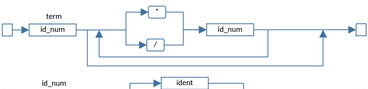
for_statement

compound_statement

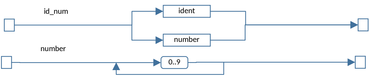
expression



term



id_num



number

ident