Page :
Date : / /

Assignment B5 B4

Title - YACC Program to validate variable declarations

Problem Statement :
Write a program using YACC specifications to implement syntax analysis phase of compiler to validate type & syntax of variable declaration in Java.

Objective :
To be proficient in writing grammar for syntax
To understand theory behind parsers
To understand automation of parsers
To be able to use YACC parsers

Outcomes : Students will be able to
Use YACC parsers
Understand concept of lex & yacc

S/W & H/W Requirement :
Working PC
64 bit Fedora OS
Text Editor
GCC Compiler
Lex Compiler
Yacc Compiler

Page:
Date: / /

Theory

YACC stands for Yet Another Compiler Compiler.

YACC is a Parser Generator which is used with lex as software tools for compiler construction.

Designing a compiler from scratch is a long & tedious task & it can be automated.
Lex & Yacc simplify the same.

Both lex & Yacc generate C - programs as output.

YACC does its work using parsing & hence it is also called a parser.

Input Descriptions :

name : names & single characters
| alternatives
;

The input of yacc is divided into 3 sections as given below

```
declaration
%%
translation rules
%%
C - functions
```

Declaration section consists of token declarations and C-code bracketed by %{ & %}

The Context Free Grammar (CFG) is placed in the rule section. Actions can be associated with each rule.

User functions are added in the last section

Commands for Yacc Assignment

↳ sudo apt-get install bison flex
↳ lex lex file.l
↳ yacc yacc-file.y
↳ cc lex.yy.c y.tab.h -ll
↳ ./a.out

Page :
Date : / /

Lex recognises regular expressions.
YACC recognises entire grammar

Lex divides input stream into tokens.
Yacc groups these tokens logically

Grammar Rules in Yacc :

The rules section of yacc defines CFG for the YACC generators, and associates with those rules C - language actions & additional precedence info.

The grammar rule has the form

A : BODY ;

A → non terminal name
BODY → sequence of zero or more names, literals, semantic actions; which are further followed by optional precedence rules.
Colon (:) & Semicolon are yacc punctuations.

Input
int a = 10
int b = 5
Conclusi

Page:
Date: / /

Input	Expected O/P	Actual O/P	Result
1) int a = 10;	int : TYPE a : IDENTIFIER = : ASSIGN 10 : NUMBER ; : SC Valid Variable Declaration	int : TYPE a : IDENTIFIER = : ASSIGN 10 : NUMBER . : SC Valid Variable Declaration	Success
2) int b = 5	int : TYPE b : IDENTIFIER = : ASSIGN 5 : NUMBER Error : Syntax Error	int : TYPE b : IDENTIFIER = : ASSIGN 5 : NUMBER Error : Syntax Error	Success

Conclusion : The program using YACC specifications to implement syntax analysis phase of compiler to validate type & syntax of variable declaration in java has been completed successfully.