Roll No: 17 TEA

7th Assignment

Nikita D Bhosale

Aim :- Design lex & yacc prog to validate type & syntax of var declaration in Java

problem statement : WAP using yacc specifies to implement lexical analysis phase of compiler to validate type and syntax of var declaration in Java

Theory :-

① yacc (yet another compiler - compiler) is a computer Program for the UNIX os developed by Steplen C Johnsen
- It is a look ahead left to right (LALR) phase to generator generating a Parsor the part of the compile that tries to make syntatic semse of the source code specifically of the LALR parser based on a analytic grammer written in the notation similar to Back us Noas form (BNF)
- yacc is supplied as a std utility based on BSP and AT and T UNIX.

② Structure of Yacc file:-
A yacc file looks much like a lex file.

```
--- defination ---
%.%.
--- rule ---
%.%.
--- Code ---
```

- As with lex all the code betn %{ and %} is copied to the begining for resulting c file
- I/P to yacc is divided in 3 sections

GURUKUL Page No.
Date

- The defination section consists of declaration and code braketed by "%{ and %}
- The BNF grammer is placed in the rules section and uses subroutine are added in subroutine sect

③ Translating, compiling & executing yacc Program

- The lex prog file consist of lex specification and should be named l. the yacc prog file consist of yacc specification and should be named y
- following command may be used to generate parser

```
lex <filename> l
yacc <filename> y
cclex.yy.c y.tab.c -ll
./a.out
```

- yacc reads the grammer descreption y and generate the parser function yy parse in file y.tabc
- The -d option causes yacc to generate the defination for tokens that are declared in the .y and placed them in file y.tab.h
- finally the lexer and the parser are compiled and linked (-ll) together to form the o/p file a.out

④ Lexical Aanalyser for yacc

- The parser and lexical analyser must agree on the token nos in order for communicate betn them

Scanned by CamScanner

- The no's may be choosen by yacc or choose by the user
- In either case the # define mechanism of c is used to allow the lexical analyser to return these no's symbolically.
- The relevant portion of the lexical analyser might look like:-

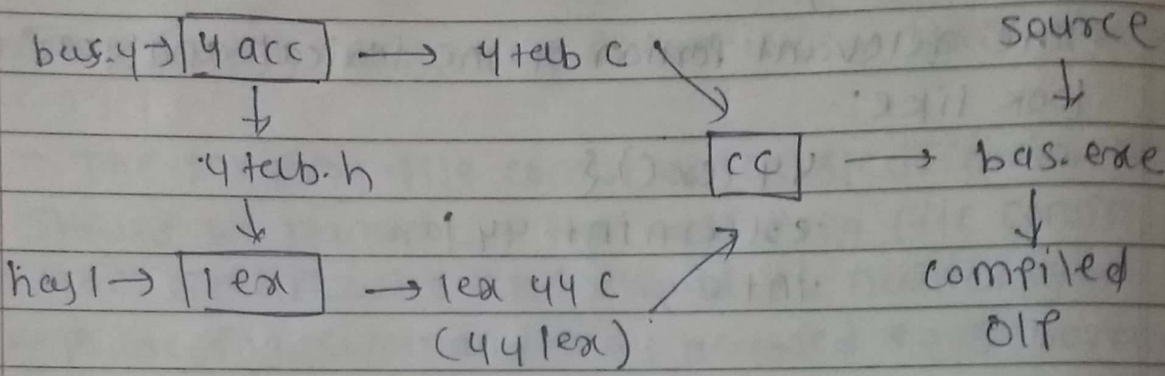
```
yylex(){
extern int yylax.
int c;
 ---
 c= getchar();
  ---switch(c){
   ---case '0';
     ---case '1':
      ----
      case '9' ;
      yylval = c-'0'
      return (DIGIT)
    ---}
```

⑤ comparing reference types:-

- sentence give structure to lang and in english they come in 4 types ~~ex~~ simple compound and complex and complex compound
- when you use several types together your writing is more interesting.
- combining sentence effectively takes practice but you'l be happy with the result

⑥ Application :-

- yacc is used to generate parser which are integral part of compiler



Conclusion :

Thus we have studied lexical analyser syntax, analysis and implemented lex & yacc application for syntax analyser to validate the given, infix expression