SRES's Sanjivani College of Engineering, Kopargaon
(*An Autonomous Institute*)
Department of Computer Engineering

SPOS Lab Manual

Assignment No. 08

AIM:
Write a program using YACC specifications to implement syntax analysis phase of compiler to recognize simple and compound sentences given in input file.

PROBLEM DEFINITION:
Write a program using YACC specifications to implement syntax analysis phase of compiler to recognize simple and compound sentences given in input file.

OBJECTIVES:
1. To understand syntax analyzer
2. To use the lex and yacc tool

INPUT:
Simple or Compound statement

OUTPUT:
To display type of statement

THEORY:
Functions used by YACC

yyparse() returns a value of 0 if the input it parses is valid according to the given grammar rules. It returns a 1 if the input is incorrect and error recovery is impossible.
yyparse() calls a routine called **yylex()** everytime it wants to obtain a token from the input.
yylex() returns a value indicating the *type* of token that has been obtained. If the token has an actual *value*, this value (or some representation of the value, for example, a pointer to a string containing the value) is returned in an external variable named **yylval**.
It is up to the user to write a **yylex()** routine that breaks the input into tokens and returns the tokens one by one to **yyparse()**.

Patterns for English Language Tokens

verb is|am|are|was|were|be|being|been|do|does|did|will|would|should|can|could|have|has|had|go|play

adverb	very|simply|gently|calmly|quietly
preposition	to|from|above|behind|below|between
conjunction	if|then|and|but|or
adjective	their|my|your|his|her|its

```
pronoun         I|you|he|she|they
noun            [a-zA-Z]+
```

Grammar for Simple and Compound Sentence

```
sentence: simple_sentence NL  {printf("Parsed a simple sentence.\n"); exit(0);}
    | compound_sentence NL  {printf("Parsed a compound sentence.\n"); exit(0);}
    ;
simple_sentence: subject verb object
    |    subject verb object prep_phrase
    ;
compound_sentence: simple_sentence CONJUNCTION simple_sentence
    |    compound_sentence CONJUNCTION simple_sentence
    ;
subject:    NOUN
    |    PRONOUN
    |    ADJECTIVE subject
    ;
verb:       VERB
    |    ADVERB VERB
    |    verb VERB
    ;
object:          NOUN
    |    ADJECTIVE object
    ;
prep_phrase:     PREPOSITION NOUN
    ;
```

ALGORITHM:

1. Write the lex program which recognise necessary tokens extension

2. Write yacc program with grammar for declarative statement

2. compile lex and yacc program to generate lex.yy.c and y.tab.c programm

3. Compile c program to generate the object program a.out

4. a.out is the generated syntax analyzer

5. Provide the input simple or Compund statement to syntax analyzer to generate the output

Executing Lex and Yacc

```
lex example.l
yacc -d example.y
gcc -o example lex.yy.c y.tab.c
./example
```

Sample Input:

He play Cricket

Output:

Parsed a simple sentence

OBSERVATION:

It is observed that above parser parse the simple or compund statement statements

CONCLUSION:

LEX and YACC tool is used to construct the parser for the simple or Compund statement statement

References:

Lex & Yacc Book by Doug Brown, John R. Levine, and Tony Mason

Prepared by
Prof.N.G.Pardeshi
Subject Teacher

Approved by
Dr. D.B.Kshirsagar
HOD