T.Y.B.Tech (CSE)

System Software and Compilers(SSC)

Lab Assignment No – 8

Name: Aniruddha Shende

Roll number: PD-05

Batch: D1

Panel: D

Name :- Aniruddha Arun Shende
Roll no :- PD-05
Batch :- D1
Panel :- D

MIT-WPU MIT WORLD PEACE UNIVERSITY | Pune

Subject :-   SSC

Lab Assignment   No - 8

Title :- Parser for Arithmetic Grammar using LEX & YACC.

Aim :- Write a program using LEX & YACC to create Parser for Arithmetic Grammar - Design calculator.

Objective :-
① To understand Yacc Tool.
② To study how to use Yacc tool for implementing Parser.
③ To understand the compilation & execution of *.y file.

Theory :- Write in brief for following :

1) Grammar - Context free grammar
Ans 1) **CFG** :- Context Free grammar is a formal grammar which is used to generate all possible strings in a given formal language.

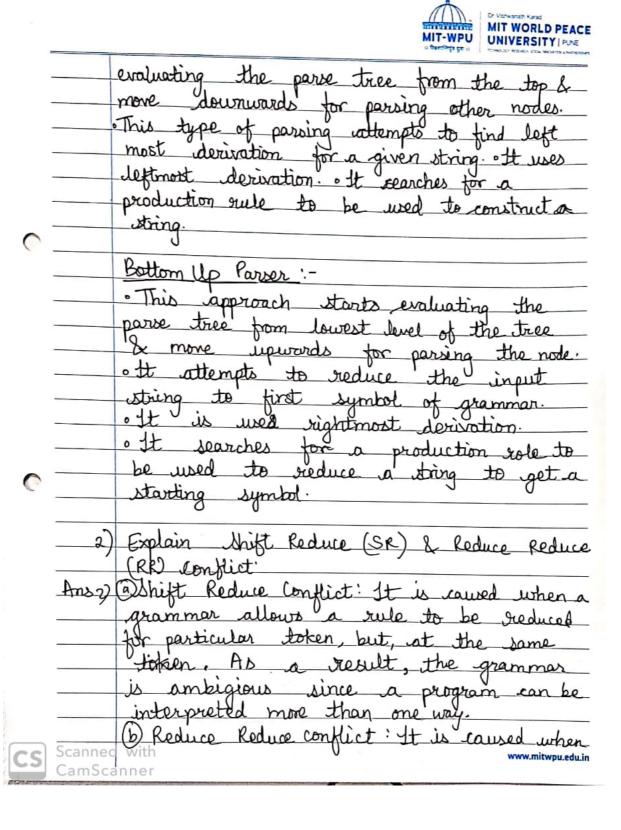2) Co routine work of Scanner & Parser

Ans 2) Most programs with flex scanners use the scanner to return a stream of tokens that are handled by a parser. Each time the program needs a token, it calls yylex(), which reads a little input and returns the token. When it needs another token, it calls yylex() again. The scanner acts as a co-routine, i.e. each time it returns, it remembers where it was, & on the next call it picks up where it left off.

3) Syntax & semantic actions in *.y file.

Ans 3) Semantic Actions describe how to compute value of attributes in tree.

Syntax actions uses:
- Regular Expression to define tokens
- Finite Automata to recognize tokens.

★ <u>Input</u> :- Source specification (*.Y & *.l) file for arithmetic expression statements.

★ <u>Output</u> :- Result of Arithmetic Expression.

FAQ's :-

1. Differentiate between top down and bottom-up parsers.

Ans 1:- <u>Top-down</u> <u>Parser</u> :- This approach starts

evaluating the parse tree from the top & move downwards for parsing other nodes.
• This type of parsing attempts to find left most derivation for a given string. • It uses leftmost derivation. • It searches for a production rule to be used to construct a string.

Bottom Up Parser :-

• This approach starts evaluating the parse tree from lowest level of the tree & move upwards for parsing the node.
• It attempts to reduce the input string to first symbol of grammar.
• It is used rightmost derivation.
• It searches for a production role to be used to reduce a string to get a starting symbol.

2) Explain Shift Reduce (SR) & Reduce Reduce (RR) conflict.

Ans2) ⓐ Shift Reduce Conflict: It is caused when a grammar allows a rule to be reduced for particular token, but, at the same token. As a result, the grammar is ambigious since a program can be interpreted more than one way.

ⓑ Reduce Reduce conflict: It is caused when

a grammar allows two or more different rules to be reduced at the same time, for the same token, when this happens, the grammar begins ambiguous since a program can be interpreted more than one way. This error can be caused when the same rule is reached by more than one path.

3) How YACC resolves ambiguias ambiguities within given grammar?

Ans 3) Shift Reduce Conflict in the parsing table is resolved by giving priority to shift move over reduce move. If string is accepted for shiftmove, then reduce move is removed, otherwise shift move is removed.

Reduce Reduce Conflict in parsing table is resolved by giving priority to first reduce move. If the string is accepted for first reduce move, then remove second, reduce move, otherwise first reduce move is removed.