Compiler Design

→ Stages of CD
→ Grammar → Unrestricted grammar
→ Regular
→ CFL (Context Free Language) ***
→ CSL (" Sensitive ")
→ Regular Exp → Grammar
→ Regular Grammar → Expression
→ Automata → DFA
→ NFA

→ Right Most Derivation
Left " "

→ Lexical Analysis
→ Syntax Analysis
→ Grammars
→ Precedence Rule
→ First & Follow
→ Parsing Table
→ LL(1) Grammar
→ Grammar → Ambiguous
→ Unambiguous

→ Handle, Item
→ Leading, Trailer
→ Operator Precedence Grammar
→ TOP-Down Parsing → Recursive
→ Install
→ Bottom-UP Parsing

→ Left Recursive Production Elimination
→ common Prefix Problem,
→ Left Factoring
→ Shift-shift conflict
→ Shift-Reduce conflict
→ Reduce-Reduce conflict
} LR(0) parser

→ LL(1) Grammar

Q) Elimination of Left Recursion:-

a) $A \rightarrow A\alpha/\beta_1/\beta_2/\beta_3$

b) $S \rightarrow (L) / a$
$L \rightarrow L, S / S$

c) $S \rightarrow A$
$A \rightarrow Ad/Ae/aB/ac$
$B \rightarrow bBc/f$

d) $X \rightarrow XSb/Sa/b$
$S \rightarrow Sb/Xa/a$