Compiler Design
Question Bank

UNIT 1

1. What is Compiler? Design the Analysis and Synthesis Model of Compiler.
2. Write down the five properties of compiler.
3. What is translator? Write down the steps to execute a program.
4. Discuss all the phases of compiler with a with a diagram.
5. Write a short note on:
   a. YACC
   b. Pass
   c. Bootstrapping
   d. LEX Compiler
   e. Tokens, Patterns and Lexemes
6. Write the steps to convert Non-Deterministic Finite Automata (NDFA) into Deterministic Finite Automata (DFA).
7. Construct Deterministic Finite Automata to accept the regular expression : (0+1)* (00+11) (0+1)*
8. Derivation and Parse Tree: Let G be a Context Free Grammar for which the production Rules are given below: Drive the string *aaabbabbba* using the above grammar (using Left Most Derivation and Right most Derivation).

   S -> aB|bA
   A -> a|aS|bAA
   B -> b|bS|aBB
9. What is Regular Expression? Write the regular expression for:
   a. R=R1+R2 (Union operation)
   b. R=R1.R2 (concatenation Operation)
   c. R=R1* (Kleen Clouser)
   d. R=R+ (Positive Clouser)
   e. Write a regular expression for a language containing strings which end with “abb” over Σ = {a,b}.
   f. Construct a regular expression for the language containing all strings having any number of a’s and b’s except the null string.

UNIT 2 AND 3

1. Explain the parsing techniques with a hierarchical diagram.

2. What are the problems associated with Top Down Parsing?

3. Write the production rules to eliminate the left recursion and left factoring problems.

4. Consider the following Grammar:
A-> ABd|Aa|a
B-> Be|b
Remove left recursion.

5. Do left factoring in the following grammar:

A-> aAB|aA|a
B-> bB|b

6. Write a short note on:
a. Ambiguity (with example)

b. Recursive Descent Parser

c. Predictive LL(1) parser (working)

d. Handle pruning

7. Write Rules to construct FIRST Function and FOLLOW Function.

8.Consider Grammar:
E-> E+T|T
T-> T*F|F
F-> (E)|id to construct FIRST Function and FOLLOW Function.

9. Write the algorithm to create Predictive parsing table with the scanning of input string.

10. Show the following Grammar:
S-> AaAb|BbBa
A-> €
B-> €
Is LL(1) and parse the input string "ba".
11. Consider the grammar:

Perform Shift Reduce Parsing for the input string using the grammar.

S->(L)|a

L->L, S|S

Input string : (a, a)
12. Write the properties of LR parser with its structure. Also explain the techniques of LR parser.

13. Write a short note on:
a. Augmented grammar

b. Kernel items

c. Rules of closure operation and goto operation

d. Rules to construct the LR(0) items
14. Consider the following grammar:
S-> Aa|bAc|Bc|bBa
A-> d
B-> d
Compute closure and goto.
15. Write the rules to construct the SLR parsing table.

16 Construct an LR parsing table for the given context-free grammar –

S–>AA

A–>aA|b

17. Write the rules to construct the LR(1) items.

18. What is LALR parser? Construct the set of LR(1) items for this grammar:
S-> CC
C-> aC

C->d

19. Show the following grammar

S->Aa|bAc|Bc|bBa
A->d
B->d
Is LR(1) but not LALR(1).

20. Write the comparison among SLR Parser, LALR parser and Canonical LR Parser.

21.Calculate FIRST and FOLLOW for the following grammar?
S-> aBDh
B-> cC
C->BC| ε
D->EF
E-> g| ε
F-> f | ε