Question Bank

UNIT I – Finite Automata

1. Define Finite Automata and Transition Diagram.
2. State the Principle of induction.
3. What is proof by contradiction?
4. Define ε-closure(q) with an example.
5. Differentiate between proof by contradiction and contrapositive.
6. Construct a DFA for the language containing “000” as a substring.
7. What is structural induction?
8. State the difference between NFA and DFA.
9. Construct DFA to recognize odd 1’s and even 0’s.
10. State the relations among RE, DFA, NFA, and NFA-ε.
11. What is inductive proof?
12. Find the set of strings accepted by the finite automata.
13. What is DFA?
14. Define epsilon transition.
15. Draw the transition diagram for an identifier.
16. What is NFA?
17. Define Deductive Proof.
18. Design DFA for strings with two consecutive 0’s.
19. What is a finite automaton?
20. Write a regular expression for strings having at least one 1.
21. Draw NFA to accept strings containing “0101”.
22. State the pumping lemma for regular languages.

23. Define languages described by DFA and NFA.
24. Define extended transition function for DFA.
25. Give RE for strings with odd number of 1’s.
26. Give RE for strings ending in 00.
27. When are two states equivalent and distinguishable?
28. What are applications of regular expressions?
29. Write REs for: (i) multiples of 2, (ii) no consecutive a’s, (iii) containing consecutive a’s.
30. State Arden’s theorem.

UNIT II – Grammars

1. Give REs for strings ending in 00 and starting with 0 and ending with 1.
2. Differentiate between RE and regular language.
3. Construct NFA for a*b*.
4. Is the regular set closed under complement?
5. Prove complement of regular language is also regular.
6. Prove 0^n1^n is not regular using pumping lemma.
7. Construct DFA for strings with (a) 4 zeros, (b) no substring 110.
8. Prove/disprove: 0^n where n is not prime is regular.
9. Construct RE for strings not containing 00 and not empty.
10. Prove closure of regular languages under concatenation and complement.
11. Give RE for strings ending in 00.
12. State pumping lemma for regular set.
13. What is a regular expression?
14. Name four closure properties of regular languages.

15. Construct NFA for (0+1)01.
16. Prove/disprove (r+s)* = r* + s*.
17. Given grammar G, analyze productions.
18. Define null and unit productions with examples.
19. Construct CFG for strings with equal number of a’s and b’s.
20. What is unambiguity?
21. Mention applications of CFG.
22. Construct CFG for palindromes.
23. What is CFG?
24. State Chomsky normal form theorem.
25. Define regular expression.
26. What is null and unit production?

UNIT III – Pushdown Automata

1. What is ambiguous grammar?
2. Define types of language accepted by PDA.
3. Specify use of CFG.
4. Define parse tree with example.
5. Construct CFG for equal number of a’s and b’s.
6. Is the language of DPDA and NPDA same?
7. Is grammar $S \rightarrow SS|(S)|S(S)S|E$ ambiguous?
8. Convert grammar to remove unit and useless productions.
9. Remove useless variables from grammar.
10. Define Pushdown Automata.

11. Write CFG for L = $\{a^n b^n\}$.
12. Is grammar $E \rightarrow E+E|id$ ambiguous? Justify.
13. What is CFG?
14. Ways of language acceptance by PDA?
15. Define PDA.
16. Describe language acceptance types by PDA.
17. Compare NFA and PDA.
18. General forms of CNF.
19. CFLs are closed under substitution?
20. Convert CFG to PDA.
21. Does PDA have memory?
22. When is PDA deterministic?
23. Notations of PDA?
24. Applications of pumping lemma in CFL.
25. Compare DPDA and NPDA.
26. Design PDA equivalent to CFG.

UNIT IV – Turing Machines

1. State pumping lemma for CFLs.
2. Applications of Turing Machine?
3. State pumping lemma for CFL.
4. What is Chomsky Normal Form?
5. Height of parse tree in CNF for string of length n?
6. Construct TM to compute n mod 2 (n in unary).

7. Design TM accepting odd integers in binary.
8. Two normal forms with examples.
9. Convert grammar to GNF.
10. Design TM accepting a(a+b)*.
11. What is a Turing Machine?
12. Fields in instantaneous description of TM?
13. Objectives of TM?
14. Is L={$a^n b^n c^n$} context free? Justify.
15. What is GNF?
16. Closure properties of CFLs.
17. Techniques for TM construction?
18. LMD/RMD for string in given grammar?
19. Define Diagonalization language Ld.
20. Define TM.
21. What is multitape TM?
22. Difference between TM and FA?
23. What is TM?
24. Multitape TM – one move explanation?
25. Applications of TM?
26. Techniques for TM construction?

UNIT V – Unsolvable Problems and Computability

1. When is a problem decidable? Give an example.
2. What is recursively enumerable language?

3. Difference between P and NP problems?
4. Prove PCP is undecidable.
5. Show PSPACE-hard is also NP-hard.
6. State Rice's Theorem.
7. Show TM set is countable.
8. Difference between decidable and undecidable problems.
9. What is Universal TM?
10. Define multiple TM.
11. Example of NP-complete problem.
12. Decidable problem and example of undecidable one.
13. What is Universal Language L_u?
14. When is REL recursive?
15. Is Tower of Hanoi tractable or intractable?
16. Define Universal TM.
17. Define NP-hard and NP-complete.
18. When is REL recursive?
19. Compare recursive and REL.
20. State decidable problem and give undecidable example.
21. Does NTM accept different language from REL?
22. Properties of REL which are undecidable.
23. Define classes P and NP.
24. When is a language recursively enumerable?
25. Define time and space complexity of TM.
26. Differentiate PCP and MPCP.