

1. What is Finite Automata?

👉 A mathematical model used to recognize regular languages.

2. Define DFA.

👉 A Finite Automaton where each input has exactly one transition.

3. Define NFA.

👉 A Finite Automaton where one input can have multiple transitions.

4. What is an input alphabet?

👉 A set of symbols used as input (denoted by  $\Sigma$ ).

5. What is a state in FA?

👉 A condition or position of the automaton.

6. What is the initial state?

👉 The starting state of FA ( $q_0$ ).

7. What is a final state?

👉 A state where input is accepted.

8. What is a transition function?

👉 It defines how states change with input.

9. Write the 5-tuple of FA.

👉  $(Q, \Sigma, \delta, q_0, F)$

10. What is  $\Sigma$  (Sigma)?

👉 Set of input symbols.

11. What is  $Q$ ?

👉 Set of all states.

12. What is  $\delta$  (delta)?

👉 Transition function.

13. What is  $q_0$ ?

👉 Initial state.

14. What is  $F$ ?

👉 Set of final states.

15. What is a transition diagram?

👉 Graphical representation of FA.

16. What is a transition table?

👉 Tabular representation of state changes.

17. What is  $\epsilon$ -transition?

👉 Transition without input symbol.

18. Which automaton allows  $\epsilon$ -moves?

👉 NFA.

19. Can DFA have  $\epsilon$ -transitions?

👉 No.

20. What type of language is accepted by FA?

👉 Regular language.

21. What is a dead state?

👉 A state from which no acceptance is possible.

22. What is a string in FA?

👉 A sequence of input symbols.

23. What is acceptance of a string?

👉 When FA reaches a final state after processing input.

24. What is determinism?

👉 Only one transition per input.

25. What is non-determinism?

👉 Multiple possible transitions for same input.

26. Can NFA be converted to DFA?

👉 Yes.

27. Which is easier to design?

👉 NFA.

28. Name one application of FA.

👉 Compiler design.

29. What is a regular language?

👉 A language accepted by FA.

30. Full form of DFA and NFA?

👉 Deterministic Finite Automata & Non-Deterministic Finite Automata.