

### Unit-I

1. A compiler is a program that:
  - A) Executes source code line by line
  - B) Converts high-level language programs into machine code
  - C) Manages computer memory
  - D) Edits source programs

**Answer: B**

2. The primary objective of a compiler is to:
  - A) Increase hardware speed
  - B) Translate source code into target code
  - C) Manage files in the operating system
  - D) Execute programs directly

**Answer: B**

3. Which of the following is NOT a phase of a compiler?
  - A) Lexical analysis
  - B) Syntax analysis
  - C) Code optimization
  - D) Program execution

**Answer: D**

4. The first phase of a compiler is:
  - A) Syntax analysis
  - B) Semantic analysis
  - C) Lexical analysis
  - D) Code generation

**Answer: C**

5. Lexical analysis is also known as:



- A) Parsing
- B) Scanning
- C) Optimization
- D) Translation

**Answer: B**

6. The output of the lexical analyzer is:

- A) Parse tree
- B) Intermediate code
- C) Sequence of tokens
- D) Object code

**Answer: C**

7. A token is:

- A) A line of source code
- B) A group of characters with a collective meaning
- C) A syntax error
- D) A machine instruction

**Answer: B**

8. Which of the following is an example of a token?

- A) Identifier
- B) Keyword
- C) Operator
- D) All of the above

**Answer: D**

9. The lexical analyzer removes:

- A) Tokens
- B) Parse trees

- C) White spaces and comments
- D) Intermediate code

**Answer: C**

10. The symbol table is used to store:

- A) Machine instructions
- B) Information about identifiers
- C) Object code
- D) Parse trees

**Answer: B**

11. Which compiler phase interacts most frequently with the symbol table?

- A) Lexical analysis
- B) Code generation
- C) Linking
- D) Loading

**Answer: A**

12. Input buffering is used in lexical analysis to:

- A) Reduce compilation errors
- B) Improve scanning efficiency
- C) Generate object code
- D) Perform optimization

**Answer: B**

13. The end-of-buffer marker is used to:

- A) Indicate syntax errors
- B) Avoid repeated buffer boundary checks
- C) Store tokens
- D) Generate parse trees

**Answer: B**

14. The two-buffer scheme in input buffering helps to:

- A) Increase memory usage
- B) Reduce input/output overhead
- C) Eliminate syntax analysis
- D) Generate code directly

**Answer: B**

15. Lex is a:

- A) Parser generator
- B) Code optimizer
- C) Lexical-analyzer generator
- D) Linker

**Answer: C**

16. The input specification for Lex is based on:

- A) Context-free grammars
- B) Regular expressions
- C) Decision trees
- D) Parse tables

**Answer: B**

17. Regular expressions are used to describe:

- A) Syntax rules
- B) Semantic rules
- C) Token patterns
- D) Parse trees

**Answer: C**

18. Which automaton recognizes regular languages?

- A) Pushdown automaton
- B) Turing machine
- C) Finite automaton
- D) Context-sensitive automaton

**Answer: C**

19. A finite automaton consists of:

- A) States and transitions
- B) Variables and constants
- C) Functions and procedures
- D) Files and records

**Answer: A**

20. An automaton with multiple transitions for the same input symbol is called:

- A) DFA
- B) NFA
- C) PDA
- D) CFG

**Answer: B**

21. In a Deterministic Finite Automaton (DFA):

- A) Multiple transitions for one input are allowed
- B) Epsilon transitions are allowed
- C) Exactly one transition exists for each input symbol
- D) No accepting states exist

**Answer: C**

22. Which of the following can contain epsilon transitions?

- A) DFA
- B) NFA

- C) Both DFA and NFA
- D) Neither DFA nor NFA

**Answer: B**

23. Converting a regular expression into an automaton usually involves creating:

- A) A parse tree directly
- B) An NFA first
- C) A symbol table
- D) Intermediate code

**Answer: B**

24. The subset construction algorithm converts:

- A) DFA to NFA
- B) NFA to DFA
- C) Grammar to parser
- D) Source code to object code

**Answer: B**

25. DFA minimization aims to:

- A) Increase the number of states
- B) Reduce the number of states
- C) Increase memory usage
- D) Generate parse trees

**Answer: B**

26. Which phase reports lexical errors?

- A) Lexical analysis
- B) Semantic analysis
- C) Code generation
- D) Optimization

**Answer: A**

27. An identifier such as “count” is recognized by the lexical analyzer as:

- A) Operator
- B) Token
- C) Statement
- D) Parse tree

**Answer: B**

28. Optimization of DFA-based pattern matchers primarily improves:

- A) Program readability
- B) Execution speed of lexical analysis
- C) Memory allocation
- D) Code generation

**Answer: B**

29. Which language construct cannot be described using regular expressions?

- A) Keywords
- B) Identifiers
- C) Nested parentheses
- D) Operators

**Answer: C**

30. The compiler phase immediately following lexical analysis is:

- A) Code generation
- B) Syntax analysis
- C) Code optimization
- D) Linking

**Answer: B**