

COMPILER DESIGN QUESTIONS WITH  
ANSWER PREPARED BY MCA DEPT. BCET

Multiple Choice Questions.

1. Compiler is a language translator but not an\_\_\_\_\_.
- A. de-translator.
  - B. preprocessor.
  - C. language migratory
  - D. assembler

ANS: B

2. The front end of a language processor analyses\_\_\_\_\_and\_\_\_\_\_.
- A. source program and semantics.
  - B. IR and syntax rule
  - C. source program and IR
  - D. semantic and lexical

ANS: B

3. The symbol table is built during\_\_\_\_\_.
- A. lexical analysis.
  - B. semantic analysis.
  - C. syntax analysis.
  - D. SP & TP analysis.

ANS: A

4. The back end in the compiler performs\_\_\_\_\_
- A. memory allocation.
  - B. code generation
  - C. memory allocation & code generation.
  - D. linking

ANS: B

5. A compiler is a \_\_\_\_\_ that reads a program written in one language, and translates it into an equivalent program in machine language.

- A. software.
- B. instruction
- C. phase
- D. program

ANS: D

6. The two parts of compilers are \_\_\_\_\_.

- A. lexical and synthesis.
- B. source program and target program.
- C. analysis and synthesis.
- D. static checkers and interpreters.

ANS: C

7. The\_\_\_\_\_part breaks up the source program into constituent pieces and creates an intermediate representation of the source program.

- A. synthesis part.

- B. analysis.
- C. syntax.
- D. argumentation.

ANS: C

8. Which is the tool that reads a program, analysis it and attempts to discover potential bugs without running the program?

- A. Structure editors.
- B. Static checkers.
- C. Interpreters
- D. Pretty pointers.

ANS: B

9. The process of streaming of character that make up the source program which is read from left- to-right and grouped into tokens is called\_\_\_\_\_.

- A. linear analysis.
- B. hierarchical.
- C. semantic analysis.
- D. streaming analysis.

ANS: A

10. The first and second basic rules of the hierarchical structure of a program are\_\_\_\_\_.

- A. any identifier or number is an expression.
- B. defining expressions in terms of operators and non recursive.
- C. if identifier1:=exp2.
- D. while (exp1) do statement2 if (exp1) then statement2.

ANS: A

11. \_\_\_\_\_are the formalization of recursive rules that can be used to guide syntactic analysis.

- A. Grammar.
- B. Context sensitive grammar.
- C. Context-free grammar.
- D. Syntax free grammar.

ANS: C

12. A \_\_\_\_\_is a data structure containing a record for each identifier, with fields for the attribute for the identifier.

- A. table.
- B. open file table.
- C. symbol table.
- D. lexical

analyzer. ANS: C

13. The \_\_\_\_\_ and \_\_\_\_\_ phases usually handle a large fraction of the errors detectable by the compilers.

- A. syntax ,semantics.
- B. semantics ,lexical.
- C. error handler ,code generator.
- D. code generator ,semantics.

ANS: A

14. The intermediate representation has a form called \_\_\_\_\_ which is like the assembly language for a machine in which every memory location can act like a register.

- A. two address code.
- B. single address code.
- C. three address code.
- D. three tier instruction.

ANS: C

15. The final phase of the compiler is the generation of target code, consisting of replaceable \_\_\_\_\_.

- A. intermediate code.
- B. semantic code.
- C. machine code
- D. buffer code

ANS: C

16. \_\_\_\_\_ is a mnemonic version of machine code.

- A. Rational preprocessors.
- B. Parse code.
- C. Intermediate code.
- D. Assembly code

ANS: D

17. The program that performs the two functions of loading and link-editing is called as \_\_\_\_\_.

- A. loaders.
- B. linkers.
- C. link-editor.
- D. optimal code.

ANS: A

18. Which one of the following is not included in front end?

- A. Semantic
- B. Syntactic
- C. Lexical.
- D. Target machine.

ANS: D

19. \_\_\_\_\_ is a process of determining if a string of tokens can be generated by a grammar.

- A. Systemization.
- B. Parsing
- C. Translation
- D. Phasing

ANS: B

20. The technique used for speeding up the lexical analyzer is \_\_\_\_\_.

- A. schemes
- B. buffer
- C. semantics
- D. sentinels

ANS: B

21. A \_\_\_\_\_ is a sequence of characters in the source program that is matched by the pattern for a token.

- A. fields
- B. table
- C. lexeme
- D. pattern

ANS: C

22. The phases of the compilers are collected into a \_\_\_\_\_ and \_\_\_\_\_ ends.

- A. primary , front.
- B. front, back.
- C. first, last.
- D. primary,

posterior. ANS: B

23. A parse tree is called as \_\_\_\_\_ syntax tree.

- A. abstract
- B. concrete
- C. comparative
- D. replicating

ANS: B

24. Consider the statement `count = count + increment;` which is the token in the statement?

- A. Count, increment.
- B. Only count.
- C. ;
- D. +

ANS: A

25. \_\_\_\_\_ is the rule describing the set of lexemes that can represent a particular token in source program.

- A. Lexeme
- B. Pattern

- C. Source
  - D. Syntax
- ANS: B

26. In the compiler model, the parser obtains a string of tokens from the \_\_\_\_\_.

- A. rest of front end.
- B. symbol table.
- C. lexical analyzer.
- D. source analyzer.

ANS: C

27. The output from the parser is usually \_\_\_\_\_.

- A. symbol table
- B. string.
- C. source programs.
- D. parse tree.

ANS: D

28. Panic mode is a type of \_\_\_\_\_ strategy.

- A. error detection.
- B. error avoidance.
- C. error recovery.
- D. error creation.

ANS: C

29. Misspelling an identifier is an example for \_\_\_\_\_ error.

- A. lexical
- B. syntactic
- C. semantic
- D. logical

ANS: A

30. If an operator is applied to an incompatible operand then it is an example for \_\_\_\_\_ error.

- A. lexical
- B. syntactic
- C. semantic
- D. logical

ANS: C

31. Usually the error detection and recovery in a compiler is done in \_\_\_\_\_ phase.

- A. lexical
- B. syntax
- C. semantic
- D. logical

ANS: B

32. Which of the following is not an error recovery strategy?

- A. Panic mode.
- B. Phrase level.
- C. Error reproduction.
- D. Global correction.

ANS: C

33. The manner in which the terminals and non terminals can be combined to form strings deals with \_\_\_\_\_ of a grammar.

- A. start symbol.
- B. forward reference.
- C. backward reference.
- D. production

ANS: D

34. Rightmost derivations are also called as \_\_\_\_\_ derivations.

- A. conical
- B. canonical
- C. triangular.
- D. depth first

ANS: B

35. A grammar that produces more than one parse tree for some sentence is said to be \_\_\_\_\_.

- A. ambiguous
- B. semantic
- C. syntactic
- D. unambiguous

ANS: A

36. An efficient non-backtracking from of top-down parser is called \_\_\_\_\_.

- A. unpredective parser.
- B. predictive parser.
- C. ambiguous parser
- D. sematic parser.

ANS: B

37. In \_\_\_\_\_ action of a shift reducer parser, the parser announces successful completion of parsing.

- A. shift
- B. reduce
- C. accept
- D. error

ANS: C

38. In the synthesis analysis model of a compiler, the front end translates a source program into an \_\_\_\_\_ code for the generation of target program.

- A. intermediate
- B. machine
- C. object
- D. lexical

ANS: A

40. Syntax tree, postfix notations and three-address codes are example for \_\_\_\_\_ language.

- A. source language.

- B. machine
- C. compiler
- D. intermediate

ANS: D

41. A linear representation of a syntax tree where it lists the nodes of a tree in which a node appears immediately after its children is \_\_\_\_\_ notation.

- A. prefix
- B. postfix
- C. infix
- D. outfix

ANS: B

42. The op field in quadruples contains \_\_\_\_\_ for the operator.

- A. external code.
- B. arguments
- C. internal code.
- D. function

ANS: C

43. Listing pointers to triples, rather than listing the triples themselves is \_\_\_\_\_ implementation.

- A. quadruples
- B. indirect triples.
- C. production
- D. start

ANS: B

44. Translating a Boolean expression into three address code without generating code for any of the Boolean operators and without hiring the code is \_\_\_\_\_ code.

- A. short circuit
- B. breaking
- C. choosing
- D. omitting

ANS: A

45. op,result,arg1 and arg2 are fields of \_\_\_\_\_.

- A. triples.
- B. indirect triples.
- C. quadruples.
- D. three address code.

ANS: C

46. An infinitively recursive call is an example for \_\_\_\_\_ error.

- A. lexical
- B. syntactic
- C. semantic
- D. logical

ANS: D

47. Elimination or replacement of unnecessary instruction in object code is called \_\_\_\_\_.

- A. code replacement
- B. code elimination
- C. code improvement
- D. code optimization

ANS: B

48. Optimization technique based on what happens across a basic block is called\_\_\_\_\_optimization.

- A. local code
- B. basic code
- C. global code
- D. lock code

ANS: C

49. Global optimizations are based on analysis.

- A. code flow
- B. data flow
- C. instruction flow
- D. program flow.

ANS: B

50. Deduction of an expression using a constant at runtime is called\_\_\_\_\_.

- A. compile time folding
- B. expression folding
- C. constant folding
- D. code folding.

ANS: C

51. The transformation of replacing an expensive operation with a cheaper one is called\_\_\_\_\_.

- A. operation reduction.
- B. operation replacement
- C. strength reduction
- D. size reduction.

ANS: C

52. The final phase in a compiler model is\_\_\_\_\_.

- A. code eliminator.
- B. code implementer
- C. code generator
- D. code optimizer.

ANS: C

53. The input of the code generator is\_\_\_\_\_.

- A. parse tree.
- B. intermediate code
- C. flow graph
- D. algorithm

ANS: B

54. The maps the IR so that the code is generated in an efficient\_\_\_\_\_manner.

- A. optimizer
- B. encoder
- C. analyzer
- D. generator

ANS: D



55. Code optimization & generation phases are often referred as\_\_\_\_\_.

- A. design phase.
- B. back end
- C. optimization phase
- D. front end.

ANS: C

56. \_\_\_\_\_selection involves choosing appropriate machine instruction to implement the IR statements.

- A. Registers
- B. Address
- C. Machine
- D. Instruction

ANS: D

57. \_\_\_\_\_selection involves what values to keep in which register.

- A. Address
- B. Value
- C. Register
- D. Instruction

ANS: C

58. \_\_\_\_\_involves in deciding the order of execution of instructions.

- A. Instruction ordering.
- B. Execution ordering
- C. Sequence ordering
- D. Code ordering.

ANS: B

59. The important criterion of a\_\_\_\_\_is to produce correct codes.

- A. code optimizer.
- B. code generator
- C. code implementer
- D. compiler

ANS: C

60. The syntax trees and DAGs are\_\_\_\_\_representations.

- A. pictorial
- B. virtual
- C. linear
- D. graphical

ANS: D

61. Postfix notations are\_\_\_\_\_representations.

- A. linear
- B. graphical
- C. virtual
- D. three-address.

ANS: A

62. The architecture of\_\_\_\_\_has a significant impact on good code generation.

- A. code generator.
- B. code optimizer

- C. target machine
- D. source machine.

ANS: A

63. Producing a relocatable machine language program is often called\_\_\_\_\_.

- A. code module
- B. program module
- C. object module
- D. data module.

ANS: B

64. The\_\_\_\_\_ must map the IR program into a code sequence that can be executed.

- A. code analyzer.
- B. code optimizer
- C. code synchronizer
- D. code generator.

ANS: D

65. \_\_\_\_\_are the fastest computational unit of a target machine.

- A. a. Compiler
- B. Analyzer
- C. Register
- D. Generator.

ANS: C

66. Flow of control can enter the basic block only through\_\_\_\_\_of the block.

- A. end instruction.
- B. first instruction
- C. middle instruction
- D. leader instruction.

ANS: A

67. The basic block becomes the mode of\_\_\_\_\_.

- A. control graph.
- B. data graph
- C. flow graph
- D. sequence graph.

ANS: B

68. The first instruction of a basic block is\_\_\_\_\_.

- A. first block.
- B. leader block
- C. header block
- D. pointer block.

ANS: D

69. Any instruction that follows conditional and unconditional jumps is\_\_\_\_\_.

- A. leader
- B. jump
- C. header

D. first  
ANS: A

70. \_\_\_\_\_ can be represented in any form of data structure that is appropriate for graphs.

- A. Control graphs.
- B. Data graphs.
- C. Coded graphs
- D. Flow graph

ANS: D

71. Many code transformations depend upon the \_\_\_\_\_ of loops in a flow graph.

- A. conditions
- B. jump
- C. execution
- D. identification

ANS: C

72. Local optimization enables improvement in \_\_\_\_\_ of code.

- A. efficiency
- B. running time
- C. execution time
- D. compile time

ANS: C

73. The important step in local optimization is transforming basic block to \_\_\_\_\_.

- A. leader.
- B. flow graph.
- C. dag.
- D. IR.

ANS:C

74. DAG representation helps to eliminate \_\_\_\_\_.

- A. data redundancy
- B. interrupts.
- C. redundant code.
- D. dead code.

ANS:D

75. \_\_\_\_\_ keeps track of the variables whose current value is in that register.

- A. Register counter
- B. Address descriptor.
- C. Stack pointer.
- D. Register

descriptor. ANS: D

76. \_\_\_\_\_ function selects the register for each associated memory location.

- A. getReg( ).
- B. ndReg( ).
- C. selReg( ).
- D. memReg( ).

ANS: A

77. A \_\_\_\_\_ is a program that converts source language into target language.

- A. compiler
- B. assembler
- C. translator
- D. analyzer

ANS: A

78. Two parts of compilation are \_\_\_\_\_.

- A. analysis and compilation.
- B. analysis and conversion.
- C. static and dynamic.
- D. analysis and synthesis.

ANS: D

79. A \_\_\_\_\_ takes as input a sequence of commands to build a source program.

- A. pretty printers.
- B. structure editors.
- C. static checkers.
- D. interpreters

ANS: B

80. \_\_\_\_\_ is called parsing.

- A. Hierarchical analysis.
- B. Semantic analysis.
- C. Synthetic analysis.
- D. Syntax tree.

ANS: A

81. A \_\_\_\_\_ is a compressed representation of parse tree in which operators appear as interior nodes and operands are children of the node.

- A. genetic tree
- B. specific tree
- C. syntax tree
- D. binary tree.

ANS: C

82. \_\_\_\_\_ are a formalization of recursive rules that can be used to guide syntactic analysis.

- A. Context free grammar.
- B. Lexical grammar.
- C. Symbol table.
- D. Semantic

analysis. ANS: A

83. Type checking is performed by \_\_\_\_\_.

- A. syntactic analysis
- B. semantic analysis.
- C. lexical analysis.
- D. schematic

analysis. ANS: B

84. Link editing is performed by \_\_\_\_\_.

- A. linker
  - B. assembler
  - C. translator
  - D. loader
- ANS: D

85. When code of one file refers to a location in another file then it is \_\_\_\_\_.

- A. internal references.
- B. external references.
- C. mutual references.
- D. explicit references.

ANS: B

86. In a compiler, \_\_\_\_\_ converts the stream of input characters into a stream of tokens

- A. syntax analyzer.
- B. semantic analyzer.
- C. type checker.
- D. lexical analyzer.

ANS: D

87. A grammar is \_\_\_\_\_ when a grammar have more than one parse tree generated with given sets of strings

- A. unambiguous
- B. absolute
- C. ambiguous
- D. reliable

ANS: C

88. A \_\_\_\_\_ is an input output mapping with syntax directed definition.

- A. notation
- B. symbol
- C. translation
- D. compilation

ANS: C

89. A parse tree showing the attribute values at each node is called \_\_\_\_\_ parse tree.

- A. inherited
- B. ambiguous
- C. production
- D. annotated

ANS: D

90. \_\_\_\_\_ is the process of determining if a string of tokens can be generated by a grammar

- A. Parsing
- B. Compiling
- C. Analyzing
- D. Translating

ANS: A

91. The term \_\_\_\_\_ denotes any finite set of symbols.

- A. strings
- B. languages

- C. alphabet
  - D. digits
- ANS: C

92. A Language denoted by a regular expression is\_\_\_\_\_.

- A. regular language.
- B. regular grammar.
- C. regular alphabet.
- D. regular set.

ANS: D

93. A deterministic finite automaton is a special case in which no state has\_\_\_\_\_.

- A. e-transition
- B. one transition.
- C. two transition.
- D. three transition.

ANS: A

94. Misspelling of an identifier falls under\_\_\_\_\_error.

- A. logical
- B. semantic
- C. syntactic
- D. lexical

ANS: D

95. \_\_\_\_\_are syntactic variable that denote set of strings

- A. Terminal.
- B. Non terminal.
- C. Notational variables.
- D. Semantic variables.

ANS: B

96. The digits 0,1,2..9 are called\_\_\_\_\_.

- A. non terminal
- B. production
- C. derivations
- D. terminals

ANS: D

97. A\_\_\_\_\_is a collection of rules for assigning type expressions to various parts of a program.

- A. system
- B. type system.
- C. semantic system.
- D. lexical system.

ANS: B

98. \_\_\_\_\_ rules govern the formation of valid statements in source language.

- A. Lexical
- B. Syntax
- C. Semantic
- D. Analysis

ANS: B

99. The \_\_\_\_\_ performs lexical, syntax and semantic analysis of SP.

- A. back end.
- B. front end.
- C. middle end.
- D. middle wall end.

ANS: B

100. In which parsing, the parser constructs the parse tree from the start symbol and transforms it into the input symbol.

- A. Bottom-up parsing
- B. Top down parsing
- C. Both A and B
- D. None of the above

ANS: B

**For any copyright issues / content error kindly drop a mail on [admin@bctorissa.org](mailto:admin@bctorissa.org)**