| Reg. No | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

# B.Tech DEGREE EXAMINATION, NOVEMBER 2023

Fifth Semester

## 18CSC362J - COMPILER DESIGN

*(For the candidates admitted during the academic year 2020 - 2021 & 2021 - 2022)*

**Note:**

i. **Part - A** should be answered in OMR sheet within first 40 minutes and OMR sheet should be handed over to hall invigilator at the end of 40th minute.

ii. **Part - B** and **Part - C** should be answered in answer booklet.

**Time: 3 Hours** **Max. Marks: 100**

**PART - A (20 × 1 = 20 Marks)**

Answer **all** Questions

| | | Marks | BL | CO |
|---|---|---|---|---|
| 1. | Keywords of a language are recognized during _____ in a compiler.<br>(A) parsing of the program (B) lexical analysis of the program<br>(C) code generation (D) dataflow analysis | 1 | 2 | 1 |
| 2. | Output of lexical analyzer is _____<br>(A) machine code (B) intermediate code<br>(C) a list of tokens (D) a parse tree | 1 | 1 | 1 |
| 3. | Major Parts of compiler are _____ in number<br>(A) 2 (B) 9<br>(C) 8 (D) 4 | 1 | 2 | 1 |
| 4. | Characters are grouped into tokens in which of the following phase of the compiler design?<br>(A) Code generator (B) Lexical analyzer<br>(C) Parser (D) Code optimization | 1 | 2 | 1 |
| 5. | In which parsing, the parser constructs the parse tree from the start symbol and transforms it into the input symbol<br>(A) Bottom-Up Parser (B) Top-Down Parser<br>(C) Both a & b (D) LR parser | 1 | 1 | 2 |
| 6. | What is the expansion of CFG?<br>(A) Compiler (B) A language expression<br>(C) Regular Expression (D) **Context Free Grammer** | 1 | 2 | 2 |
| 7. | A context free language is called ambiguous if ----------?<br>(A) It has 2 or more than 2 left derivations for some terminal string w ϵ L (G) (B) It has 2 or more than 2 right derivations for some terminal string w ϵ L (G)<br>(C) It has 2 or more than 2 left and right derivations for some terminal string w ϵ L (G) (D) It has 2 or more than 2 left derivations | 1 | 2 | 2 |
| 8. | Grammar that can be translated to DFA's is _____<br>(A) Left linear grammar (B) Right linear grammar<br>(C) Generic grammar (D) linear grammar | 1 | 2 | 2 |
| 9. | A bottom up parser generates _____<br>(A) Right most derivation (B) Rightmost derivation in reverse<br>(C) Leftmost derivation (D) Leftmost derivation in reverse | 1 | 2 | 3 |

10. Shift Reduce parsers are _____ 1 2 3
(A) Top down Parser (B) Bottom Up parser
(C) May be top down or bottom up (D) LR Parser

11. Which of these is also known as look-head LR parser? 1 2 3
(A) SLR (B) LR
(C) LLR (D) Top down Parser

12. What is the similarity between LR, LALR and SLR? 1 2 3
(A) Use same algorithm, but different parsing table (B) Same parsing table, but different algorithm
(C) Their Parsing tables and algorithm are similar but uses top down approach. (D) Both Parsing tables and algorithm are different

13. A grammar that produces more than one parse tree for some sentence is called _____ 1 2 4
(A) Ambiguous (B) Unambiguous
(C) Regular (D) Both a and b

14. Which of the following is true for machine language? 1 2 4
(A) Continuous execution of program segments (B) Depicting flow of data in a system
(C) A sequence of instructions which solves a problem (D) The language which interacts with the computer using only the binary digits 1 and 0

15. The average time required to reach a storage location in memory and obtain its contents is called the _____ 1 2 5
(A) Seek time (B) Turnaround time
(C) Access time (D) Transfer time

16. Which loader function is accomplished by loader? 1 2 5
(A) Reallocation (B) Allocation
(C) Linking (D) Loading

17. Which of the following are storage allocation strategies 1 2 5
(A) Static allocation (B) Stack allocation
(C) Heap allocation (D) Both a and b

18. DAG is an abbreviation of? 1 1 6
(A) Detecting Acyclic Graph (B) Data Acyclic Graph
(C) Dynamic Acyclic Graph (D) **Directed Acyclic Graph**

19. The graph that shows basic blocks and their successor relationship is called _____. 1 2 6
(A) DAG (B) Flow graph
(C) control graph (D) Hamiltonion graph

20. Which is not part of runtime memory subdivision? 1 1 6
(A) Stack (B) Heap
(C) Static data (D) Access link

## PART - B (5 × 4 = 20 Marks)

Marks BL CO

Answer **any 5** Questions

21. Design a Deterministic Finite Automata (DFA) to accept strings having 101 as a substring over $\Sigma=\{0,1\}$. Write the formal definition of the DFA and draw the transition table. Also show that the string 110100 is accepted by the DFA. 4 1 1

22. Analyze and elaborate on the various stages of the compiler, with a neat diagram. 4 2 1

| | | | | |
|---|---|---|---|---|
| 23. | Find the Leftmost and Rightmost derivation and draw tree for given expression<br>$E \rightarrow E + E \mid E * E \mid (E) \mid a \mid b \mid c$<br>$\omega = id + id * id$ | 4 | 2 | 2 |
| 24. | Give the rules for First and Follow and for the given grammar identify<br>i. First ()<br>ii. Follow () | 4 | 2 | 2 |
| 25. | Construct the precedence table for the following grammar. Determine the Lead and Trail.<br>$E \rightarrow EAE$<br>$A \rightarrow +\mid *$ | 4 | 2 | 3 |
| 26. | Evaluate the expressions for the SDD annotated parse tree for the follow expressions. 3 * 5 + 4n. | 4 | 1 | 4 |
| 27. | List and elucidate any three issues in the design of a code generator. | 4 | 2 | 5 |

**PART - C (5 × 12 = 60 Marks)** — **Marks** **BL** **CO**

Answer **all** Questions

| | | | | |
|---|---|---|---|---|
| 28. | (a) With a neat diagram, describe the *Compiler construction tools and Input Buffering.* | 12 | 2 | 1 |
| | **(OR)** | | | |
| | (b) Explain in detail with a neat diagram<br>i) Analysis Phase ii) Synthesis Phase | | | |
| 29. | (a) Construct a Predictive LL(1) parser and match the input id+id*id is matching with the given grammar.<br>E--> TE`<br>E--> +TE`<br>T--> FT`<br>T`--> *FT`/ϵ<br>F--> (E) /id | 12 | 3 | 2 |
| | **(OR)** | | | |
| | (b) Construct stack Implementation of shift reduce parsing for the grammar<br>$S \rightarrow (L) \mid a$<br>$L \rightarrow L, S \mid a$<br>Show whether the following string will be accepted or not. (a, ((a, a), (a, a))) | | | |
| 30. | (a) Explain in detail about DAG for basic blocks with example. | 12 | 3 | 4 |
| | **(OR)** | | | |
| | (b) Display the parse tree in top-down translation scheme for following input string real, id1, id2, id3. | | | |
| 31. | (a) Construct Stack Implementation of Shift Reduce parsing for the grammar<br>E->E+E<br>E->E*E<br>E->(E)<br>E->id and the input string id1+id2*id3. | 12 | 4 | 3 |
| | **(OR)** | | | |
| | (b) Explain the flow of control statements for Boolean expression. | | | |
| 32. | (a) Discuss the various peephole optimization techniques in detail. | 12 | 4 | 5 |
| | **(OR)** | | | |
| | (b) Elaborate the features and algorithm for DAG and construct the dag for the following basic block:<br>d: = b * c<br>e: = a + b<br>b: = b * c<br>a: = e – d | | | |

* * * * *