(6 pages)

Reg. No. : .....

Code No. : 6438 **Sub. Code : ZCSM 14**

M.Sc. (CBCS) DEGREE EXAMINATION, NOVEMBER 2022

First Semester

Computer Science – Core

COMPILER DESIGN

(For those who joined in July 2021 onwards)

Time : Three hours Maximum : 75 marks

PART A — (10 × 1 = 10 marks)

Answer ALL questions.

Choose the correct answer :

1. The intermediate code can be directly executed using a program called a
   (a) Complier (b) Interpreter
   (c) Macro (d) Scanner

2. In a transition diagram, the states are connected by arrows called
   (a) Labels (b) Letters
   (c) Design (d) Edges

3. The syntactic specification of a programming, language, use a notation called
   (a) Context-free grammer
   (b) Regular Expression
   (c) Syntax grammer
   (d) Regular grammer

4. The bottom-up parsing method is called
   (a) shift-reduce parsing
   (b) canonical derivations
   (c) rightmost derivations
   (d) recursive descent parsing

5. A tree in which each leaf represents an operand and each interior node an operator is called
   (a) Parse tree (b) Heap tree
   (c) Code tree (d) Syntax tree

6. Which statement is an abstract from of intermediate code?
   (a) 3-address (b) 2-address
   (c) 1-address (d) address

7. The instruction MOV R0, R1 implies.

(a) copies R1 into R0 (b) moves R1 into R0

(c) copies R0 into R1 (d) moves R0 into R1

8. When the value of a variable is changing at each step in a loop called

(a) undefined variable

(b) undeclared variable

(c) uninitialized variable

(d) Induction variable

9. Redundant loads and stores results in

(a) efficient run

(b) wasted time and space

(c) good algorithm

(d) efficient programming

10. Conditional statements are used in

(a) Program (b) global registers

(c) loops (d) global variables

PART B — (5 × 5 = 25 marks)

Answer ALL questions, choosing either (a) or (b).
Each answer should not exceed 250 words.

11. (a) Discuss about Language Processors.

Or

(b) Explain about Nondeterministic Finite Automata.

12. (a) Write down the Formal Definition of a Context-Free Grammer.

Or

(b) Write down the algorithm to eliminate left recursion from a grammar.

13. (a) Categorize the Three-Address Code.

Or

(b) Classify Translation of Expressions.

14. (a) Summarize the DAG Representation of Basic Blocks.

Or

(b) Evaluate Addresses in the Target Code.

15. (a) Express the Principal Sources of Optimization.

Or

(b) Generalize Loops in Flow Graphs.

PART C — (5 × 8 = 40 marks)

Answer ALL questions, choosing either (a) or (b)
Each answer should not exceed 600 words

16. (a) Illustrate the Specification of Tokens.

Or

(b) Summarize the design of a Lexical-Analyzer Generator.

17. (a) Examine the Bottom-Up Parsing.

Or

(b) Determine the Stack Allocation of Space.

18. (a) Classify Backpatching.

Or

(b) Analyze the Unification algorithm.

19. (a) Generalize the Code-Generation Algorithm.

Or

(b) Justify Peephole Optimization.

20. (a) Evaluate he Lazy-Code-Motion Algorithm.

Or

(b) Test the Data-Flow Analysis.

———————