Reg. No. : | | | | | | | | | | | | |

Question Paper Code : 81174

B.E./B.Tech. DEGREE EXAMINATIONS, NOVEMBER/DECEMBER 2025.

Fifth Semester

Computer Science and Engineering

CS 3501 — COMPILER DESIGN

(Regulations 2021)

Time : Three hours | Maximum : 100 marks

Answer ALL questions.

PART A — ($10 \times 2 = 20$ marks)

1. Differentiate tokens, patterns and lexeme.

2. List down the differences between compiler and Interpreter.

3. Distinguish between SLR, CLR, LALR parsers.

4. List down the Problems in Top Down Parsing.

5. What are the three functions of backpatching?

6. Differentiate between L attribute and S attribute.

7. What are the issues in the design of a code generator?

8. List down the Operations of Symbol table.

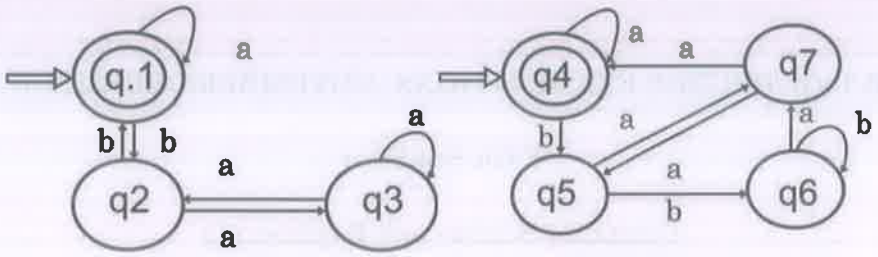
9. What is the Role of peephole optimization in compilation process?

10. List down the applications of DAG.

PART B — (5 × 13 = 65 marks)

11. (a) (i) Describe the role of lexical analysis in compiler design. (6)

(ii) Check for the equivalence of the two Automata given below. (7)



AUTOMATON-1 AUTOMATON-2

Or

(b) (i) Explain LEX Tool with a Lex Programme. (9)

(ii) Check whether the following grammar is ambiguous or not. (4)

$S \rightarrow AB$
$A \rightarrow Ca$
$B \rightarrow Ba / b$
$C \rightarrow Cb / b$

12. (a) (i) Define augmented grammar. Construct the LR(0) items for the following Grammar. (10)

$S \rightarrow L = R$
$S \rightarrow R$
$L \rightarrow *R$
$L \rightarrow id$
$R \rightarrow L$

(ii) List down the Features of YACC. (3)

Or

(b) (i) Consider the grammar.

$S \rightarrow xABC$
$A \rightarrow a \mid bbD$
$B \rightarrow a \mid \varepsilon$
$C \rightarrow b \mid \varepsilon$
$D \rightarrow c \mid \varepsilon$

Construct predictive parsing table for the given grammar. (9)

(ii) List down the Features of Syntax Analysis. (4)

13. (a) Explain in detail about Back patching Technique.

Or

(b) Explain different types of intermediate code representations.

14. (a) (i) Explain Storage allocation strategies with suitable examples. (9)

(ii) Write three address code for the expression (4)

$a+a*(b-c)+(b-c)*d$

Or

(b) (i) Write about all issues in code generation. Describe it. (7)

(ii) Compare Static allocation, stack allocation and heap allocation with their merits and limitations. (6)

15. (a) (i) Explain optimization techniques on Basic Blocks with simple examples. (9)

(ii) Define Dead-code elimination with example. (4)

Or

(b) (i) Explain the peephole optimization technique. (9)

(ii) What is control and data flow analysis? Explain with example. (4)

PART C — (1 × 15 = 15 marks)

16. (a) (i) Construct the DAG for following statement. $a+b*c+d+b*c$. (8)

(ii) Consider the following grammar (7)

$E \rightarrow E+T/T$
$T \rightarrow TF/F$
$F \rightarrow F*/a/b$

Construct SLR parsing table for the above grammar.

Or

(b) (i) Explain the simple code generator and generate target code sequence for the following statement $d:=(a-b)+(a-c)+(a-c)$. (8)

(ii) Convert the regular expression $abb(a/b)*$ to DFA. (7)