

K20P 0558

Reg. No. : .....

Name : .....

**IV Semester M.C.A. (Including Lateral Entry Stream) Degree (C.B.S.S. – Reg./  
Supple./Imp.) Examination, May 2020  
(2014 Admission Onwards)**

**MCA 4C21 : SYSTEM PROGRAMMING AND COMPILER DESIGN**

Time : 3 Hours

Max. Marks : 80

**SECTION – A**

Answer any ten questions. Each question carries three marks. (10×3=30)

1. Write a note on advanced macro facilities.
2. Enlist the steps for assembler design.
3. Write a note two-pass assembler.
4. Write a note on loaders.
5. With an example, explain regular expressions.
6. Write a note on specification of tokens.
7. Write a short note on YACC.
8. Write a note on predictive parser.
9. Write a note on bottom up translation.
10. Write a note on synthesized attributes.
11. Convert the expression :  
 $a = b^* - c + b^* - c$   
into Three Address statements ?
12. Write a note on runtime storage management.

P.T.O.



## SECTION – B

Answer all questions. Each question carries ten marks. (5×10=50)

13. a) Explain two-pass macro processor with flowchart and databases. 10  
 b) Explain in detail design of a macro pre-processor. 10
14. a) What are the phases in the design of a compiler ? Show the output of the compiler phases for an input string position : = initial + rate + 60. Indicate atleast one error that can be detected at each phase of the compiler. 10  
 b) Explain in detail design of a lexical analyzer generator. 10
15. a) Construct SLR table for 10  
 $S \rightarrow BB$   
 $B \rightarrow b B \mid d$   
 And also find the following input is valid ?  
 bbddb
- b) i) Explain the error recovery in LR parsing. (5+5)  
 ii) Define left recursion. Is the following grammar left recursive ?  
 $E \rightarrow E + E \mid E * E \mid a \mid b$
16. a) Explain the different storage allocation strategies in detail. 10  
 b) i) Write a note on syntax-directed definitions. (5+5)  
 ii) Explain in detail specification of a simple type checker.
17. a) What are the issues in design of a code generator ? Explain in detail. 10  
 b) Explain principal sources of code optimization techniques with example. 10