

**MCA-15**  
December - Examination 2015  
**MCA IInd Year Examination**  
**System Programming**  
**Paper - MCA-15**

**Time : 3 Hours ]**

**[ Max. Marks :- 80**

---

**Note:** The question paper is divided into three sections A, B and C. Write answers as per given instructions.

**Section - A**

8 x 2 = 16

(Very Short Answer Questions)

**Note:** Answer all questions. As per the nature of the question delimit your answer in one word, one sentence or maximum upto 30 words. Each question carries 2 marks.

- 1) (i) What is overlays?
- (ii) Define macro.
- (iii) What is POOL?
- (iv) What you mean by translator?
- (v) What is OP CODE? Give example.
- (vi) Write three steps of Divide and conquer algorithm.
- (vii) List the drawbacks of recursive parser.
- (viii) What is Bootstrapping?

**Section - B**

4 x 8 = 32

(Short Answer Questions)

**Note:** Answer any four questions. Each answer should not exceed 200 words. Each question carries 8 marks.

- 2) Compare Scanning and Parsing with example.
- 3) Describe various optimizing transformations commonly used in compilers.
- 4) Convert given regular expression to DFA. The expression is  $(a | b)^*abb\#$
- 5) Give example(s) of errors detected by first three phases of compiler.
- 6) Explain the basic features of Assembly Language Programming.
- 7) How literal are different from constant? Explain.
- 8) What is the function of interpreter? Why it is used?
- 9) Explain left factoring by giving example.

**Section - C**

2 x 16 = 32

(Long Answer Questions)

**Note:** Answer any two questions. You have to delimit your each answer maximum upto 500 words. Each question carries 16 marks.

- 10) List various phases of a language processor. Explain roles of first two phases of it. Also explain symbol table.
- 11) Explain and compare various intermediate code forms (representations) for an assembler.

- 12) What is the role of parser? Compare Top-Down Parsing with Bottom-Up Parsing.
  - 13) Discuss the advantages of DFA. Draw a DFA for bit strings with at least one '0' and at least one '1'.
-