

**MCA-15**  
December - Examination 2016  
**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 × 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) Give differences between compiler and interpreter.
- (ii) Why code optimization is used?
- (iii) Name the different phases of a compiler.
- (iv) What is memory allocation?
- (v) What do you mean by parsing?
- (vi) Define LR grammar.
- (vii) Define NFA.
- (viii) Define absolute loader.

**Section - B****4 × 8 = 32**

(Short Answer Questions)

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

- 2) What is the purpose of Symbol table?
- 3) What are the different kinds of assembly language statement? Explain.
- 4) Define loader? What are the functions of loader?
- 5) Why code optimization is used? How it is achieved?
- 6) Differentiate between top-down parsing and bottom-up parsing.
- 7) Define compiler. Mention the different phases of a compiler.
- 8) What do you mean by parsing? Define parse tree. Note down the role of parser in compiler design.
- 9) Define loading, relocating, relative loader and linking?

**Section - C****2 × 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) Discuss the functions of both phases of an assembler during translating a source program to its processed form.
- 11) Briefly explain a simple bootstrap loader, with an algorithm show the design of a bootstrap loader.
- 12) What is memory allocation? Discuss static and dynamic memory allocation during compilation of a program.
- 13) Draw a NFA that recognize the language of all strings that end in aab.