Roll No…… Total Pages: 3

10003

MCA/M-17

PRINCIPLES OF PROGRAMMING LANGUAGE

Paper: MCA-14-23

Time: Three Hours Maximum Marks: 80

Note: Attempt five questions including No. 1 which is compulsory. All questions carry equal marks.

Compulsory Question

1. (a) Comment on the role of programming languages.
   (b) What do you mean by type checking?
   (c) What is ambiguous grammar?
   (d) What do you understand by parsing?
   (e) Differentiate between classes and modules.
   (f) What do you mean by implicit sequence control?
   (g) Name the various ways to store variables.
   (h) What is an applet?

Unit-I

2. (a) State and explain the features of a good programming language.

   (b) What do you understand by binding? How can you classify bindings? Explain with suitable examples.

2. What is a recursive-descent parser? Explain by using suitable examples. Also explain the various characteristics of a grammar that allows a recursive-descent parser.

Unit-II

4. (a) Show that the S-aSb|abS|V grammar is ambiguous, and find an equivalent unambiguous grammar.

   (b) Explain about context-free-grammers with examples.

5. (a) What do you understand by data type? Describe the specification and implementation of data types in various languages?

   (b) What is the role of Determinism in Pushdown automata? Give an example of a deterministic pushdown automata accepted languages with PDA ?

Unit-III

6. (a) What are the various ways that software component should be modified for reuse ?

   (b) Describe various implementation issues in Object-Oriented languages.

7. (a) Discuss the various problems with evaluation of expressions.

   (b) How the subprogram sequence control is handled? Explain.

Unit-IV

8. (a) Describe the statement-level, procedure-level and program-level parallelism in parallel programming.

   (b) What is network programming? Explain various features and objectives of network programming.

9. What are the various programmer and system controller storage managements? Explian static and heap storage management in detail using appropriate examples.