MCA-14-23 PRINCIPLES OF PROGRAMMING LANGUAGES

Maximum marks: 100 (External: 80, Internal: 20) Time: 3 hours

Note: Examiner will be required to set NINE questions in all. Question Number 1 will consist of objective type/short-answer type questions covering the entire syllabus. In addition to question no. 1, the examiner is required to set eight more questions selecting two from each unit. Student will be required to attempt FIVE questions in all. Question Number 1 will be compulsory. In addition to compulsory question, student will have to attempt four more questions selecting one question from each Unit. All questions will carry equal marks.

UNIT - I

Preliminaries: History, Impact of Programming Paradigms, Role of Programming Languages, Good Language, Effects of Programming Environment, Translators and virtual architectures, Binding and Binding time, Language Syntax, Analysis of Program, Synthesis of Object program, Formal translation models: BNF Grammars, General parsing, Language translation, Recursive descent parsing.

UNIT - II

Formal languages and automata: The Chomsky hierarchy of formal languages, regular grammars, Regular expressions, Finite State Automata, Context-free grammars, Pushdown automata, Ambiguous grammars.

Language Semantics: Attribute grammars, Denotational semantics, Program verification and validation, Data objects, variables, constants, data types, declaration, type checking, type casting, type promotion, Enumerators, Composite data types.

UNIT - III

Object Orientated concepts: Structured data types, Abstract data types, Information hiding, Subprogram concepts, Good program design, Type definitions, Type equivalence, Inheritance, Derived classes, Abstract classes, Polymorphism, Inheritance and software reuse.

Sequence control: Implicit and explicit sequence control, Sequence control within arithmetic expressions, sequence control between statements, sequencing with non-arithmetic expressions, Subprogram Sequence control.

UNIT - IV

Miscellaneous topics: Parameter passing techniques, Static & Dynamic Scoping, Storage of variables, Static storage, Heap Storage management, Distributed Processing, Exceptions and Exception handlers, Coroutines, Scheduled subprograms, Parallel programming, Processor design, Hardware and Software architectures, Network Programming, Evolution of scripting languages, Applets, XML.

Text Books:

1. Pratt T.W., Zelkowitz M.V., Gopal T.V., “Programming Languages Design and Implementation”, Pearson Education.
2. Sebesta W. Robert, “Concepts of Programming Languages”, Pearson Education.

Reference Books:

1. Appleby Doris & VandeKopple J. Julius, “Programming languages-Paradigm and practice”, Tata McGraw Hill.
2. Sethi Ravi, “Programming languages”, Pearson Education
3. Scott M., “Programming Language Pragmatics”, Elsevier India.