

Course No: MCA – 1T3
Course Title: Programming Language Principles

Unit I

The role of Programming Languages : Towards Higher Level Languages programming paradigms , Language implementation. Language Description : Syntactic Structures , Expression Notations , Abstract Syntax trees , Lexical Syntax , Context free grammars , grammars for expression. Imperative Programming : Structured Programming , Syntax directed control flow , Design considerations , handling special cases in loops , programming with invariants , proof rules for partial correctness , control flow in C

Unit II

Data Representation : The role of types , basic types , arrays , records , unions and variant records , Sets , Pointers , Two String tables , types and error checking. Procedure Activations : Introduction to Procedures , parameter passing methods, scope rules for names , nested scope in source text, activation records ,lexical scope : procedures as in C

Object oriented programming : Constructs for program structuring , Information hiding , Program design and modules , modules and defined types , class declarations in C++ , dynamic allocation in C++ , templates : Parameterized types , Implementation of Objects in C+., Inheritance , derived classes and information hiding

Unit III

Functional Programming : Language of expressions , types, values and operations , approaches to expression evaluation , lexical scope , type checking, Function declaration by cases , Functions as first-class values ,Implicit types , data types exception handling , Scheme , a dialect of Lisp , the structure of lists , list manipulation, Simplification of expressions.Logic Programming , Computing with relations , Introduction to Prolog , data structures in Prolog , Programming techniques , controls in Prolog, Cuts

Unit IV

, An introduction to concurrent Programming : Parallelism in hardware , Streams : implicit synchronization , concurrency as interleaving, Liveness properties , safe accesses to shared data concurrency in ADA.

Language Description : Semantic Methods , Synthesized attributes, Attribute grammars , natural semantics , Denotational Semantics , Equality of Pure Lambda terms , Substitution revisited , Computation with pure lambda terms , programming constructs as lambda terms , the typed lambda calculus , polymorphic types

Reference Books:

1. RaviSethi,“ Programming Languages ,Concepts and Constructs”, Pearson Education
2. Freidman, Wand ,Haynes, ”Essentials of Programming Languages”, PHI.