UE18CS321: PRINCIPLES OF PROGRAMMING LANGUAGES (4-0-0-0-4)

# of Hours: 56

Class #	Chapter Title/Reference Literature	Topics to be Covered	% of Portions Covered	
			% covered	Cumulative
1.	**Unit: I** **Preliminary Concepts; Names, Binding, Type Checking and Scopes;** **Chapter 1 and Chapter 5**	**Preliminaries:** Reasons for studying concepts of programming languages, Programming domains	21.4%	21.4%
2.		Language Evaluation Criteria		
3.		Influences on Language design, Language categories, Programming Paradigms		
4.		Programming Language Implementation – Compilation and Virtual Machines		
5.		**Names, Binding, Type Checking and Scopes:** Names, Variables, Type bindings, Type Inferencing, Type Checking, Strong Typing.		
6.		**Case Study**: Linux utilities and Program Debuggers for languages such as C, Python.		
7.	**Unit: II** **Type Checking and Scopes, Data types:** **Chapter 6, Chapter 7 - 7.6, 7.7, 7.8**	**Type Checking and Scopes (continued..):** Type Equivalence, Scope and Lifetime, Referencing Environments.	21.4%	42.8%
8.		**Data types:** Introduction, primitive, character, user defined, array, associative		
9.		record, union, pointer and reference types, design and implementation issues related to these types.		
10.		Names, Variables, concept of binding, type checking,		
11.		type compatibility, named constants, variable initialization.		
12.		**Expressions and Statements:** Short circuit evaluation mixed mode assignment, Assignment Statements.		

13.	**Unit: III** **Control Structures, Subprograms and Blocks** **Chapter 8, 9**	**Control Structures**: Statement Level, Compound Statements, Selection, Iteration	21.4%	64.2%
14.		Unconditional Statements, and guarded commands.		
15.		**Subprograms and Blocks:** Fundamentals of sub-programs, Scope and lifetime of variable		
16.		static and dynamic scope		
17.		Design issues of subprograms and operations, local referencing environments, parameter passing methods,		
18.		overloaded sub-programs, generic sub-programs, parameters that are sub-program names		
19.	**Unit: IV** **Functions, Abstract Data Types, Object Oriented Concepts** **Chapter 9.11, 9.12, 9.13,** **Chapter 11, 12.**	**Functions** (continued..)**:** Design issues for functions, user defined overloaded operators, co routines and Function closures.	17.8%	82.1%
20.		**Abstract Data types:** Abstractions and encapsulation, introduction to data abstraction, design issues.		
21.		Object oriented concepts.		
22.		Object oriented concepts. (Continued...)		
23.		Object oriented concepts. . (Continued...)		
24.	**Unit: V** **Exception Handling, Logic Programming and Functional Programming** **Chapter 13, 14, 15 and 16**	**Exception handling:** Exceptions, Specifications, Exception Propagation.	17.8%	100%
25.		**Logic Programming Language:** Introduction and overview of logic programming,		
26.		Basic elements of prolog, application of logic programming.		
27.		**Functional Programming Languages:** Introduction, fundamentals of FPL,		
28.		Application of Functional Programming Languages and exploration of the features.		
(Note: Each class is of 2 Hour duration.)				

Literature

Book Type	Code	Title & Author	Publication Information		
			Edition	Publisher	Year
Text Book	T1	Concepts of Programming Languages, Robert .W. Sebesta	10th	Pearson Education	2012
Reference Book	R1	Programming Language Pragmatics, Michael L. Scott	3rd	Elsevier	2009
	R2	Programming Languages Design and Implementation – Pratt and Zelkowitz	4th	PHI/Pearson Education	2001