

**Course: Major
Paradigm of Programming Languages**

Semester: I	Credits: 4	Subject Code: SMAJCPPL123552	Lectures: 60
--------------------	-------------------	-------------------------------------	---------------------

Course Outcomes:

At the end of this course, the learner will be able to:

- CO1-Learn major programming paradigms and techniques involved in design and implementation of modern programming languages.
- CO2-Design and develop programs using the Scala programming language.
- CO3-Analyze methodologies, design/implementation issues involved with variable allocation and binding with respect to various programming languages.
- CO4-Understand the evolution of data types and subroutines.
- CO5- Understand the concept of object orientation.
- CO6 -Understand the concept of concurrent programming.

Unit 1: Introduction to the programming languages

15

- Introduction to the programming languages- History and need of various types of programming languages (PL), Types of programming languages, Characteristics of programming languages, Language Evaluation Criteria, Syntax, Semantics, Pragmatics Analysis of programming languages.
- Functional Programming in Scala-Strings, Numbers, Classes and properties, Methods, Objects, Functional Programming, List, Array, Map, Set.

Unit 2: Scope, Control flow in programming languages

15

- Scope -The Notion of Binding Time, Object Lifetime and Storage Management, Static Allocation, Stack-Based Allocation, Heap-Based Allocation, Garbage Collection, Scope Rules, Static Scoping, Nested Subroutines, declaration order, Dynamic Scoping, meaning of Names in a Scope, Aliases, Overloading, Polymorphism and binding of Referencing Environments, Subroutine Closures, First-Class Values and Unlimited Extent, Object Closures, Macro Expansion.
- Control flow-Expression Evaluation- Precedence and Associativity, Assignments, Initialization, ordering within Expressions, Short-Circuit Evaluation, Structured and Unstructured Flow-Structured Alternatives go to, Sequencing-Selection Short-Circuited Conditions, Case/Switch Statements, Iteration -Enumeration-Controlled Loops, Combination Loops, Iterators, Controlled Loops, Recursion, Applicative and Normal-Order evaluation.

Unit 3: Data types and Subroutines

15

- Datatypes-Primitive Data Types, Numeric Types-Integer, Floating point, Complex, Decimal, Boolean Types, Character Types, Character String Types, Design Issues, Strings and Their Operations, String Length Operations, Evaluation, Implementation of String types,
- User defined Ordinal types -Enumeration types, implementation of user defined ordinal types.
- Array types-Design issues, Arrays and indices, Subscript bindings and array categories, Heterogeneous arrays, Array initialization, Array operations, Rectangular and Jagged arrays, Slices, Evaluation, Implementation of Array Types,

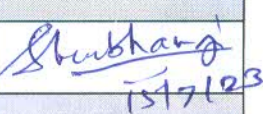
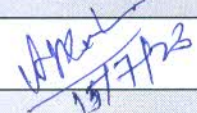
Board of Studies	Department	Name	Signature
Chairperson (HoD)	B.Sc(Comp. Sci.)	Ashwini Kulkani	<i>Ashwini</i> 11/11/23




<p>Associative Arrays & implementation.</p> <ul style="list-style-type: none"> ● Record type- Definitions of records, References to record fields, Operations on records, Evaluation, Implementation of Record types ● Pointer and Reference Types- Design issues, Pointer operations, Pointer problems, Dangling pointers, Solution to dangling pointer problem, Lost heap dynamic, variables, Comparison of Pointers in C and C++, Referencetypes, Evaluation, Implementation of pointer and reference types. ● Subroutines- Fundamentals of Subprograms, design Issues for subprograms, Local Referencing Environments, parameter, Passing Methods, Parameters that are Subprograms, Overloaded Subprograms, design Issues for Functions, User-Defined Overloaded Operators, Coroutines, General Semantics of Calls and Returns, Implementing Subprograms with Stack-Dynamic Local Variables, Nested Subprograms, Blocks, Dynamic scope. 	
--	--


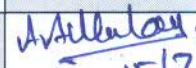

<p>Unit 4: Object Orientation and Concurrency</p> <ul style="list-style-type: none"> ● Object-Oriented Programming- Encapsulation and Inheritance, Initialization and Finalization, Choosing a Constructor, References and Values, Execution Order, Garbage Collection, Dynamic Method Binding, Virtual- and Non-Virtual Methods, Abstract Classes, Member Lookup, Polymorphism, Object Closures, Multiple Inheritance ● Introduction to concurrent programming- Introduction to Concurrency, categories of concurrency, Subprogram-level, concurrency Fundamental concepts, Language design for concurrency, design Issues, Semaphores, Monitors, Introduction to Message Passing, concept of Synchronous Message Passing JavaThreads, The Thread class, Priorities 	15
---	----

<p>Reference Books:</p> <ul style="list-style-type: none"> ● Alvin Alexander, <i>Scala Cookbook</i>, O'Reilly Publication ● Elsevier, <i>Scott Programming Language Pragmatics</i>, Kaufmann Publishers ISBN 9788131222560. ● Robert W. Sebesta, <i>Concepts of Programming</i>, Pearson Education
--

Board of Studies	Name	Signature
Chairperson (HoD)	Mrs. Ashwini Kulkarni	
Faculty	Mrs. Swati Pulate	
Faculty	Mrs. Smita Borkar	
Faculty	Mrs. Shubhangi Jagtap	
Faculty	Mrs. Alka Kalhapure	
Faculty	Mrs. Anjali Kale	
Subject Expert (Outside SPPU)	Dr. Aniket Nagane	
Subject Expert (Outside SPPU)	Dr. Manisha Divate	

Board of Studies	Department	Name	Signature
Chairperson (HoD)	B.Sc(Comp. Sci.)	Ashwini Kulkani	



VC Nominee (SPPU)	Dr. Reena Bharathi	 15/7/23	
Industry Expert	Ms. Anjali Ingole		 15/7/23
Alumni	Ms. Pooja Pandey	 15-8-23	



Board of Studies	Department	Name	Signature
Chairperson (HoD)	B.Sc(Comp. Sci.)	Ashwini Kulkani	