Course Code	21CSC203P	Course Name	Advanced Programming Practice			Course Category	P	Professional Core		L	T	P	C
										2	1	0	3

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil

Course Offering Department	Computational Intelligence	Data Book / Codes/Standards	Nil

Course Learning Rationale (CLR): The purpose of learning this course is to:

		Learning			Program Outcomes (PO)														
CLR-1	Understand the paradigm functionalities and their hierarchy	1	2	3	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CLR-2	Knowledge on structural, procedural, and Object-Oriented Programming Paradigm																		
CLR-3	Proficiency in event, Graphical User Interface, and declarative Paradigm with a java application.																		
CLR-4	Extended knowledge on logic, functional, network and concurrent Paradigm																		
CLR-5	Symbolic, Automata-based, and Event with a python application.																		

Course Outcomes (CO): (CO): At the end of this course, learners will be able to:

CO	Description	Level of Thinking (Bloom)	Expected Proficiency (%)	Expected Attainment (%)	Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual & TeamWork	Communication	Project Mgt. & Finance	LifeLong Learning	PSO-1	PSO-2	PSO-3
CO-1	Devise solutions using various programming paradigm	2	75	70	3	2	-	-	-	-	-	-	-	-	-	-	-	2	-
CO-2	Express proficiency in the usage of structural, procedural, and Object-Oriented Program	4	75	70	3	2	-	1	-	-	-	-	-	-	-	-	-	2	-
CO-3	Develop Java application using declarative, event, and graphical user interface paradigm	6	75	70	3	-	2	-	2	-	-	-	-	1	-	-	-	2	-
CO-4	Express proficiency in the usage of logic, functional, network, and concurrent Paradigm	4	75	70	3	2	-	1	-	-	-	-	-	-	-	-	-	2	-
CO-5	Design and develop Python application using symbolic, automata-based, and graphical user interface programming paradigms	6	75	70	3	-	2	1	2	-	-	-	-	1	-	-	-	2	-

Unit 1 – INTRODUCTION TO PROGRAMMING PARADIGM
Programming Languages – Elements of Programming languages - Programming Language Theory - Bohm- Jacopini structured program theorem - Multiple Programming Paradigm – Programming Paradigm hierarchy – Imperative Paradigm: Procedural, Object-Oriented and Parallel processing – Declarative programming paradigm: Logic, Functional and Database processing - Machine Codes – Procedural and Object-Oriented Programming – Suitability of Multiple paradigms in the programming language - Subroutine, method call overhead and Dynamic memory allocation for message and object storage - Dynamically dispatched message calls and direct procedure call overheads – Object Serialization – parallel Computing.

Unit 2 – JAVA PROGRAMMING PARADIGMS
Object and Classes; Constructor; Data types; Variables; Modifier and Operators - Structural Programming Paradigm: Branching, Iteration, Decision making, and Arrays - Procedural Programming Paradigm: Characteristics; Function Definition; Function Declaration and Calling; Function Arguments - Object-Oriented Programming Paradigm: Abstraction; Encapsulation; Inheritance; Polymorphism; Overriding - Interfaces: Declaring, Implementing; Extended and Tagging - Package: Package Creation.

Unit 3 – ADVANCED JAVA PROGRAMMING PARADIGMS
Concurrent Programming Paradigm: Multithreading and Multitasking; Thread classes and methods - Declarative Programming Paradigm: Java Database Connectivity (JDBC); Connectivity with MySQL – Query Execution; - Graphical User Interface Based Programming Paradigm: Java Applet: Basics and Java Swing: Model View Controller (MVC) and Widgets; Develop a java project dissertation based on the programming paradigm.

Unit 4 – PYTHONIC PROGRAMMING PARADIGM
Functional Programming Paradigm: Concepts; Pure Function and Built-in Higher-Order Functions; Logic Programming Paradigm: Structures, Logic, and Control; Parallel Programming Paradigm: Shared and Distributed memory; Multi-Processing – Ipython; Network Programming Paradigm: Socket; Socket Types; Creation and Configuration of Sockets in TCP / UDP – Client / Server Model.

Unit 5 – FORMAL AND SYMBOLIC PROGRAMMING PARADIGM
Automata Based programming Paradigm: Finite Automata – DFA and NFA; Implementing using Automaton Library - Symbolic Programming Paradigm: Algebraic manipulations and calculus; Sympy Library - Event Programming Paradigm: Event Handler; Trigger functions and Events – Tkinter Library. Develop a python-based project dissertation based on the programming paradigm.

Learning Resources	1. Elad Shalom, A Review of Programming Paradigms throughout the History: With a suggestion Toward a Future Approach, Kindle Edition, 2018 2. Maurizio Gabbrielli , Simone Martini, Programming Languages: Principles and Paradigms, 2010.	3. Herbert Schildt, Java: The Complete Reference Seventh Edition, 2016. 4. Mark Lutz, Programming Python: Powerful Object-Oriented Programming, 2011.

Learning Assessment

Bloom'sLevel of Thinking		Continuous Learning Assessment (CLA) - By the Course Faculty						By The CoE	
		CLA-1 Average of Unit test (20%)		CLA-2 Project Based Learning (60%)		Report and Viva Voce (20% Weightage)		Final Examination (0% weightage)	
		Theory	Practice	Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	30	-	--	20	-	10	-	-
Level 2	Understand	30	-	-	20	-	10	-	-
Level 3	Apply	20	-	-	20	-	10	-	-
Level 4	Analyze	20	-	-	20	-	10	-	-
Level 5	Evaluate	-	-	-	10	-	30	-	-
Level 6	Create	-	-	-	10	-	30	-	-
Total		100 %		100 %		100 %		-	

CLA – 4 can be from any combination of these: Assignments, Seminars, Tech Talks, Mini-Projects, Case-Studies, Self-Study, MOOCs, Certifications, Conf. Paper etc.

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
Mr. N. Venkatesh, Tech Lead, Honeywell, Bengaluru, Karnataka, India	Dr. Sudeepta Mishra, Assistant Professor, Computer Science and Engineering, Indian Institute of Information Technology, Ropar, Punjab.	Dr Ramkumar J, Assistant Professor, Computing Technologies, SRM Institute of Science and Technology, KTR Campus, Chennai